Experimental Feedback Interfaces for Consumer Activity Tracking Wearable Devices

by

Christopher Noel Lloyd

B.A. Philosophy & Psychology
Wheaton College, 2009

Submitted to the Integrated Design and Management Program
in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE IN ENGINEERING AND MANAGEMENT
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

MAY 2020

Signature of Author: ______________________________

Christopher Lloyd
Integrated Design and Management Program
May 21, 2020

Certified by: ______________________________

Maria Yang
Professor
Department of Mechanical Engineering
Thesis Supervisor

Accepted by: ______________________________

Matthew S. Kressy
Director
Integrated Design and Management Program

This page is left intentionally blank

Experimental Feedback Interfaces for
Consumer Activity Tracking Wearable Devices

by

Christopher Noel Lloyd

Submitted to the Integrated Design and Management Program
in partial fulfillment of the requirements of the degree of Masters of
Science in Engineering and Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2020

Abstract

Commercial wearable activity trackers have sophisticated monitoring capabilities and digital user interfaces that report personal health metrics; however, these devices have not yet achieved their goal of dramatically improving the wellbeing and performance of users. This research identifies latent aspects of wearables that might improve wellbeing. A group of commercial wearable users are interviewed to determine unmet and latent needs. Qualitative interview data is leveraged to propose a case study of a flower robot as a figurative feedback interface that uses moving mechanisms to express the user's sleep quality and promote improved sleeping habits. The robotic flower user interface is divided into two components that are fabricated and tested separately: 1) a flower that blooms and 2) a stem that changes posture. The control system is fabricated, programmed, and tested to successfully retrieve the researcher's personal sleep data from a public API and actuate the stem and flower. The flower robot prototype is a proof of concept of a novel commercial activity tracking wearable interface. Further testing is required to determine if a robotic avatar can increase relevant task performance, change user behavior change, or improve health metrics..

Thesis Supervisor: Maria Yang

Title: Professor, Department of Mechanical Engineering

This page is left intentionally blank

Acknowledgements

I thank my thesis advisor and mentor, Professor Maria Yang, for taking me on as a teaching assistant and guiding my research for this thesis. Her scholarship and lectures on prototype fidelity became invaluable as the Covid-19 crisis shut down campus workshops and makerspaces and I had to adjust my research. I must also acknowledge Professor Neil Gershenfeld for showing someone with a liberal arts background that he actually could make (almost) anything. The work I did in his course became the basis of my thesis.

I wouldn't be in graduate school if not for my partner, Jason Scott, who pushed me to explore my academic interest in design and apply to MIT, even though he dreaded leaving California. Through the unbearable cold, he has remained a steadfast encourager and fellow brainstormer. I thank Celene Reynolds, PhD, and Professors Kareem Khubchandani, Nikki Yeboah, AB Brown, Mbongeni Mtshali, and Kemi Adeyemi, friends who all charted the path of graduate education before me. Their foresight and comic relief cut through the roadblocks on my journey to complete my studies.

I thank the Integrated Design and Management Program, Tony Hu, Andy MacInnis, and Matthew Kressy for inviting me to be a part of the IDM community. I will cherish the relationships and memories I have formed with my cohort for a lifetime.

This page is left intentionally blank

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Historical context

Wearable heart rate monitors began in the 1980s with medical and sports science applications. Later, activity trackers became available to the public that incorporated accelerometers and altimeters as an input to detect movement and gave way to what consumers know as activity trackers, fitness wearables, or just “wearables,” as they are commonly known. New algorithms have been developed and applied to sensor data to detect unexpected activities such as sleep and wakefulness phases (Grandnera and Rosenberger, 2019). Consumer wearables for tracking health and wellness data take on many forms, including watches, wrist bands, skin patches, badges, jewelry, smart garments, eyewear, shoe insoles, and headbands with embedded sensors that are connected to a mobile or tablet application designed to monitor physical and psychological stress, provide cognitive or physical feedback (biofeedback), and/or monitor and promote sleep (Peake et al, 2018).

Smart wristwear products remain the most popular among mass market consumers for tracking health, with Apple and Fitbit taking the leading share of the market (Statista, 2019). Additionally, the smart watch and wristband markets are expected to grow from 137.5 million units sold in 2019 to 219.3 million units sold in 2020 (International Data Corporation [IDC], 2020). Many wrist-based products also include digital interfaces that mix health applications with other functionality such as tracking time, notifications, messaging and games.

1.2 Product architecture

A review of wearable technology sensors for consumer sports applications finds a common architecture as seen in Figure 1 (Aroganam et al, 2019). A simplified summary proceeds as follows: 1) Sensors monitor user activity, 2) smartphones or tablets receive sensor data using Bluetooth or Bluetooth Low Energy (BLE) broadcast, 3) data is then transmitted to a cloud server through Wifi or cellular transmission where the signal is processed and data is analyzed, and 4) the data is then transmitted back to the smartphone and subsequently transmitted to the wearable (if it has an interface) to be displayed to the user. Some sensor signal processing and analysis occurs natively in the wearable device and smartphone application as well. User input survey input is received in addition to sensor input through the wearable or smart phone interface.

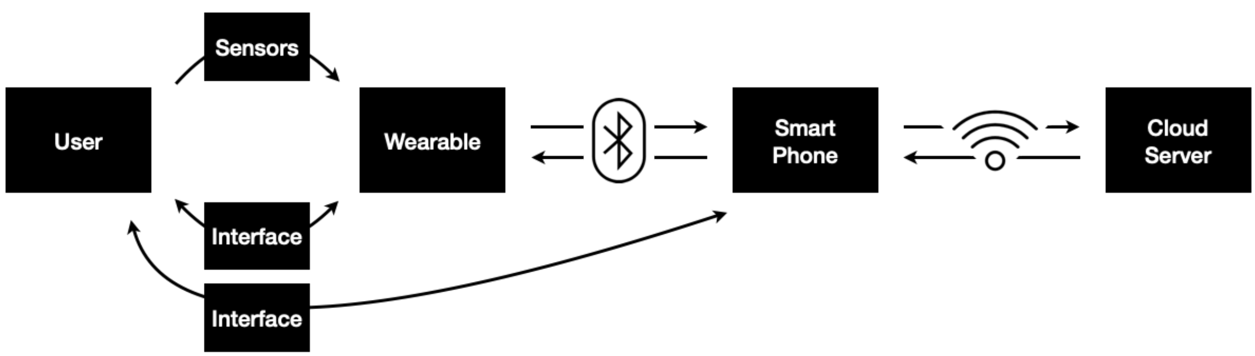


Figure 1: Consumer wearable product architecture adopted from Aroganam et al (2019)

1.3 Sensors and data

The first generation of consumer wearable sensors returned simple metrics such as counting steps of monitoring heart beats per minute. As sensor technology improves and becomes more reliable, these trackers are able to monitor a wider range of inputs, including blood oxygen levels, breathing rate, muscle electrical activity, stress/emotion, cognitive function, movement patterns, sweat analysis, and sleep architecture (Peake et al, 2018). And as signal processing algorithms improve, products are able to deliver more accurate and sophisticated insights that go beyond high-level metrics. For example, a sensor that tracks a user's heart rate may be able to also produce data that tracks sleep and wakefulness phases (like REM, slow wave sleep and light sleep), analysis that previously required an electroencephalogram (EEG) test (Bandyopadhyay et al, 2019).

Sensor data is also shared to more outlets than ever before. Many consumer wearable companies have taken a platform business model approach and allow users to share data with third parties. Sensor data can be broadcast via Bluetooth to nearby devices, such as a stationary bicycle that displays heart rate. Sensor data can be shared through API services to third-party applications that provide additional reporting insights and services.

1.4 User interfaces

As consumer wearable hardware and software expanded to new capabilities and applications, firms primarily focused on optimizing the mobile application user experience. Dashboards that were once viewed on websites or desktop applications are now optimized for mobile phones. Detailed graphs and dashboards are more likely to be consolidated into a high level score or simplified graphic. As the data collected becomes more sophisticated, however, the same amount of visual real estate is available on a smartphone interface.

1.5 Motivational affordances

The stated purpose of wearable businesses is to help their users “live a healthier, more active life” (Fitbit, 2020). Almost all mobile applications connected to consumer wearables use some form of motivational affordances, or “gamification,” to achieve this mission. A survey of 50 fitness and health applications found that 97% of applications used gamification techniques to promote physical activity or weight loss (Cotton & Patel, 2018). Weiser et al (2015) developed a taxonomy of motivational affordances that include the following building blocks:

- Assignments, quests, goals
- Points, credits, levels
- Achievements, badges
- Leaderboards, collections
- Reminders
- Virtual goods
- Friends, groups, teams

A review of the home screens of Fitbit and Apple Health, the mobile applications for the two most popular consumer wearable applications, found that 1) Assignments, quests, and goals, 2) reminders, and 3) friends, groups, and teams were most prominently featured.

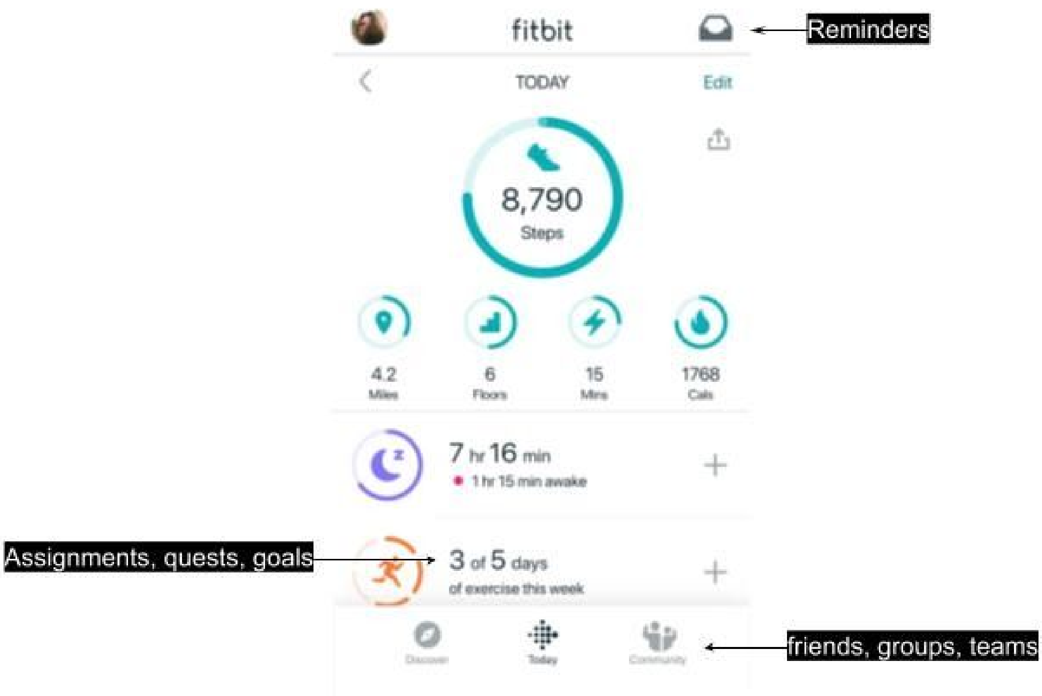


Figure 2: Home screen display of Fitbit smartphone application (2020)

1.6 Analytical vs. figurative design

Mobile applications for wearable activity trackers exclusively display user data in analytical dashboards, but other models of data displays exist. Figurative design uses visual metaphors instead of scores, metrics or text to display feedback. An example of this distinction can be seen in figure 3: a water faucet digital display as designed by Bao et al (2018).

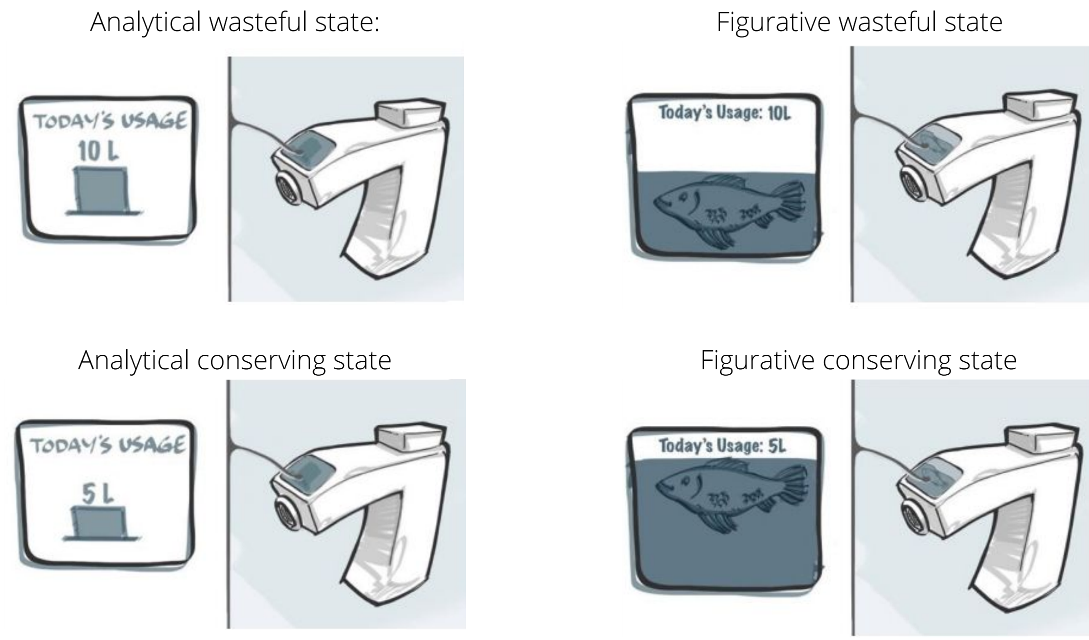


Figure 3: Example of analytical and figurative feedback interfaces (Bao et al, 2018)

Bao et al conducted a study on the emotional response to "eco-feedback" designs, comparing an analytical design and a figurative design for resource conservation products like the one in figure 3. An analytical feedback interface displays the number of liters of water used in a certain time period. A figurative display might display a fish in water that lowers as water is used—suggesting that high water usage will kill the fish. Kahneman's (2013) model of system 1 (fast and automatic) and system 2 (slow and deliberate) thinking would suggest that the figurative interface may be more salient to a user because the dying fish is an emotional visual story that taps into system 1 thinking, whereas the analytical interface may require an effortful system 2 response to process the information.

Opportunity exists to use figurative expression of wearable sensor data as a form of motivational affordance. A classic example of a product with figurative design is the Tamagotchi pet.

Figure 4: Tamagotchi Pet (Wikipedia, 2019)

Tamagotchi was incredibly popular, not only because of motivational affordances that reminded and rewarded the user to care for and grow his or her demanding Tamagotchi pet, but also because of the emotional relationship developed between young users and the digital avatar. Bao's research on eco-feedback specifically found that younger subjects have a more significant emotional response to figurative feedback over analytical feedback.

1.7 Impact on behavior

It can be difficult to tell if consumer wearables make impactful behavior and lifestyle change or simply satisfy personal curiosity. Several studies using randomization have shown that wearable usage increases physical activity (Wang, et al., 2016, Ashe, et al., 2015, Chung, et al., 2016). However these studies have limited sample sizes with durations ranging from one day to six months. Among the research reviewed, Jakicic et al (2017) ran the largest (n = 237) randomized study for two years on individuals participating in a behavioral weight loss intervention and found that the technology-enhanced intervention made no difference in outcomes.

1.8 Opportunity selection

This thesis proposes to identify and explore an opportunity to increase behavior change and lifestyle improvements among individuals who use consumer wearables. While new sensors and data analysis have unlocked domains of personal health and wellness insight, visual feedback displays have not advanced as quickly. This thesis will develop a case study that explores how an experimental interface can display activity tracker data in intuitive ways.

Using Terwiesch and Ulrich's (2009) three horizon model helps us understand opportunities in the changing landscape of technologies and consumer behavior. This model uses two axes: 1) knowledge of user need and 2) knowledge of technology to identify three horizons of innovation opportunities.

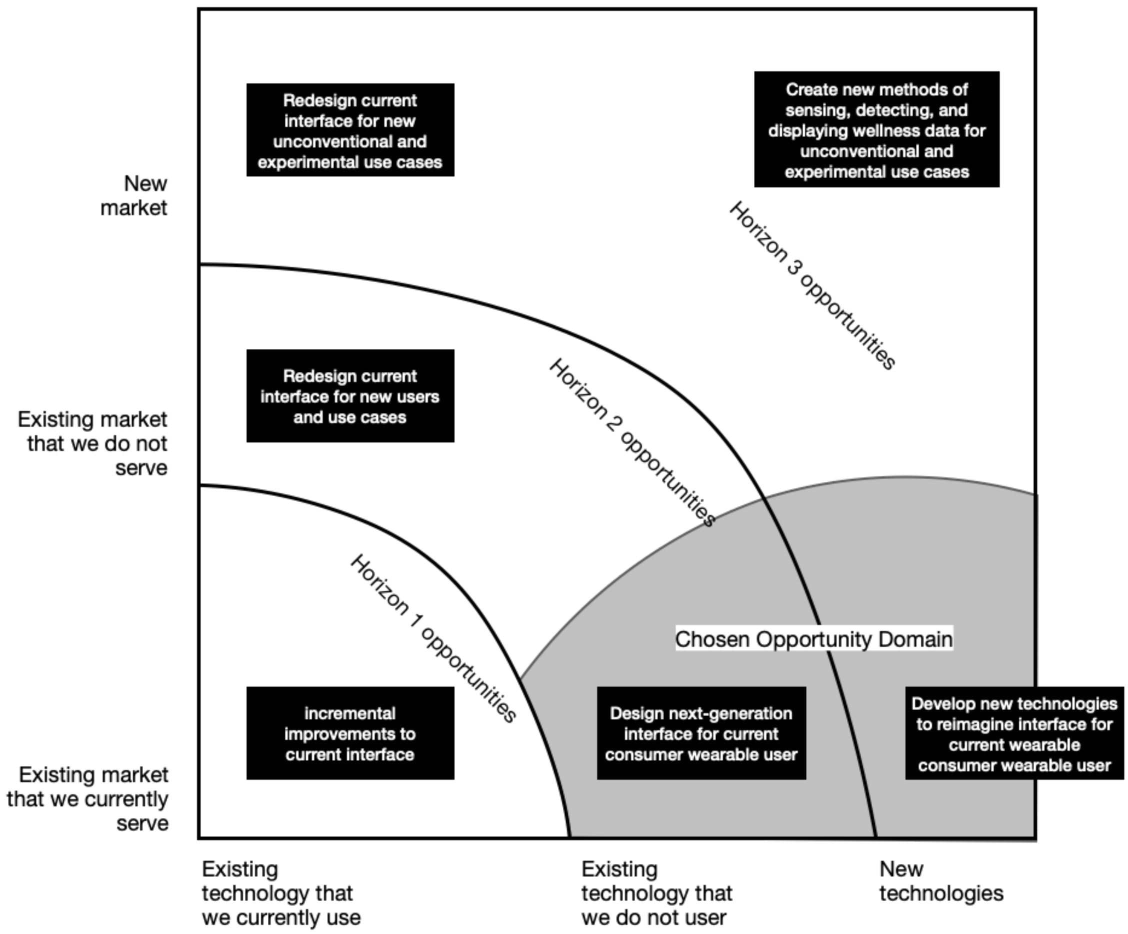


Figure 5: Three horizons diagram adapted from Terwiesch and Ulrich (2009)

Based on the levels of opportunity within the consumer wearable domain, this thesis will focus on Horizon 2 and 3 opportunities to create an experimental interface. Because the platform of wearable technologies is already well established as a category, the exploration for this thesis will focus on the lower right quadrant of Horizon 3 opportunities, which means exploring new applications of technology to address the existing market of consumer wearable users.

1.9 Approach to explore opportunity

This thesis will explore the opportunity of experimental interfaces for consumer wearables using the human centered design approach as taught in the MIT Integrated Design and Management (IDM) curriculum. This method follows the lineage of Rolf Faste, who developed literature and practice design processes at Stanford University in the 1980s. The following diagram articulated by Camburn et al (2017) demonstrates the four stages commonly followed in this design innovation process.

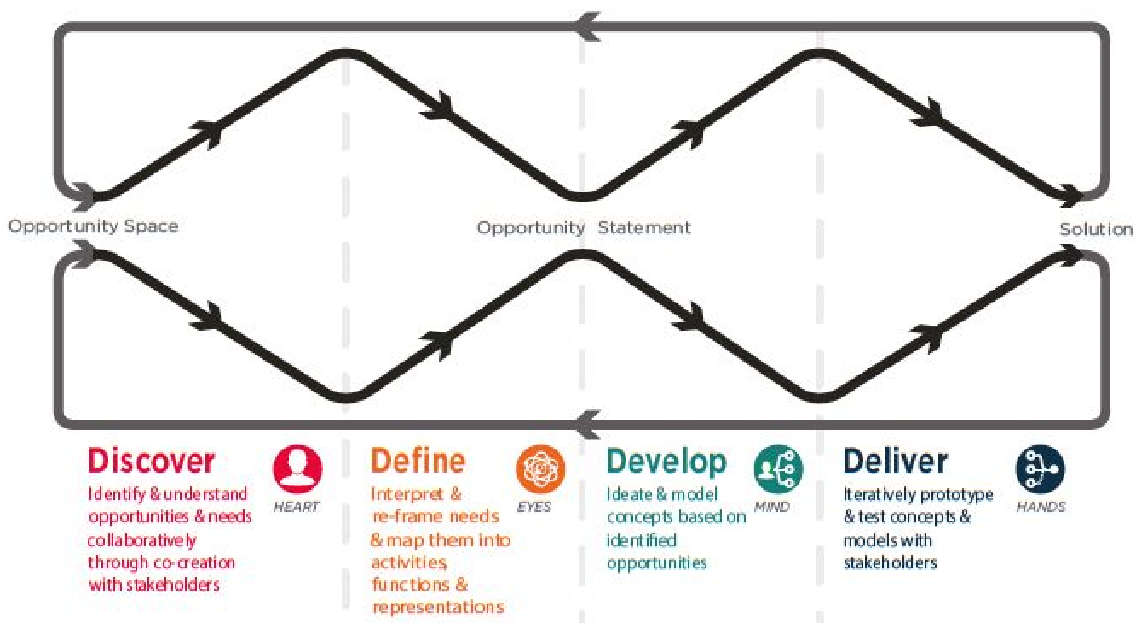


Figure 6: Four stages of design innovation process (Camburn et al, 2017)

Applying the methodology as described by Camburn et al (2017) in Fig. 7, this thesis follows the following four steps:

1. Discover – Understand the needs of current consumer wearable users through interviews, participant observation and empathy building.

2. Define - Document and frame those needs.
3. Develop - Generate and select promising concepts that meet those needs.
4. Deliver - create and test prototypes to determine the viability, desirability and feasibility of the concept and iterate as needed.

Chapter 2: Design Research

2.1 Research approach

In order to develop an interface that motivates individuals to adopt behaviors that promote balanced wellness, one must first learn from end-users to understand strengths and weaknesses of current interfaces (Ulrich et al, 2019). This form of user interview is also known as the articulated-use method (Camburn et al, 2017), which seeks to elicit how products, services or systems are used and can be improved. Collected data is synthesized and articulated by the researchers into user needs.

2.2 Participant selection

Seven participants are selected to participate in one-hour interviews. Research suggests that over 80% of user needs can be uncovered within the first five one-on-one interviews (Griffin and Hauser, 1993; Ulrich et al, 2019).

Interview selection protocol:

- A brief screening survey was sent to several outlets at a university in the northeast region of the United States, including the following programs:
    - Students and recent alumni of a graduate design program via Whatsapp channel: 124 members
    - Students in graduate business administration program via Slack channel: 1,823 members

- Participants were asked basic contact information, which activity tracking wearable(s) that they own, the frequency of use, and what metrics were most important to them.
- Participants were selected who use a wearable activity tracker multiple times per week or every day and expressed interest in tracking multiple types of data (activities, sleep, heart rate, etc). The rationale for this selection is to better understand users who are actively trying to improve their overall health and wellbeing. In total, seven participants were selected for interviews.

Participant	Age Range	Gender	Wearable Used
1	25-29	Female	Fitbit
2	30-34	Male	WHOOP
3	30-34	Female	Lintelek
4	25-29	Male	Apple Watch
5	25-29	Female	Garmin
6	35-39	Female	Fitbit
7	25-29	Female	Apple Watch

Table 1: Interview participant list

The invitation pool, comprised of current and recent students of professional graduate programs, limited the age range of selected interviewees (exclusively ages 25-34). However, this age range represents the largest 10-year age segment of consumer wearable owners—34.9% of the market, according to the Statista Global Consumer Survey (2019). That same survey found that wearable usage is split evenly between male and female consumers, while this study skewed towards female users. Additional research could be done to determine if user needs differ by gender, but these differences are not explored in this project.

2.3 Interview protocol

Interviewees were invited to participate in a one-hour video call. Basic demographic information was collected and each participant received the following questions:

- What type of activity tracker do you use?
- What type(s) of data do you use? Which metrics are most important to you?
- Why do you track this data?
- How does seeing this data change your behavior, if at all?
- Walk me through how you used your activity tracker today (or yesterday). What worked well, what wasn't useful?
- How do you define "wellbeing?"
- If you could change one thing about your lifestyle to promote your wellbeing, what would it be?

These questions reveal how activity trackers are being used and understand explicit and implicit needs that are not being addressed by the current product interface. In some cases, interviewees demonstrated product usage and screenshots of wearable interfaces. Relevant statements and observations were anonymously captured and put into a database to be assigned themes as they related to other interviewee statements.

2.4 Defining user needs

The researcher interpreted each interviewee's need statements after re-listening to the interview for a second time (Faste, 1987). Each user need statement was assigned to the following three categories:

a. an explicit need currently met by the existing product,
b. an explicit need that isn't met by the existing product, and
c. a latent need that the interviewee may not be aware of themselves.

After all interviews were transcribed, each interview resulted in approximately 7-8 core user needs statements. These statements were organized by theme using an affinity grouping exercise as seen in Figure 7.

Some interesting themes also emerged among multiple interviewees:

- Several interviewees discussed budgeting sleep and activities throughout the day, suggesting there isn't one metric that is most important to them, but balancing multiple metrics.
- Several interviewees look at complex datapoints, such as sleep and wakefulness phases (duration of time in light, slow wave and REM sleep); however, they expressed confusion about how to actually use the data or constructed theories about how sleep cycles affect their day based on anecdotal experience.
- Many interviewees expressed the desire to reach a "good enough" state as opposed to peak performance. This suggests a different category of use case than the traditional fitness and weight loss applications of activity trackers.

Figure 7: Affinity Diagram of User Needs

2.5 Prioritizing user needs

Four interviewees responded to an invitation to return for a follow-up interview to help determine the relative importance of the user needs. Griffin & Hauser (1993) demonstrates that the frequency of mentioning a need in user interviews does not correlate with the level of importance of that need. Therefore, additional feedback is required to determine the relative importance of each need. Understanding the relative weight of user needs can also help make design trade-off decisions later in the development process (Clausing, 1998). Each follow-up interviewee was invited to participate in a 15-minute video call using Zoom. The prioritization exercise took place using Mural, an application that allows the interviewer and interviewee to collaborate on the same digital canvas as seen in figure 8..

Interview protocol:

- Each participant is read a short introduction that explains the purpose of the study and the exercise they will participate in [see figure 8].
- In each session, user needs are randomly ordered in a 3 X 5 block..
- The interviewee is asked to order the user needs on a spectrum from least important to most important.
- The interviewee is encouraged to think out loud and ask questions.

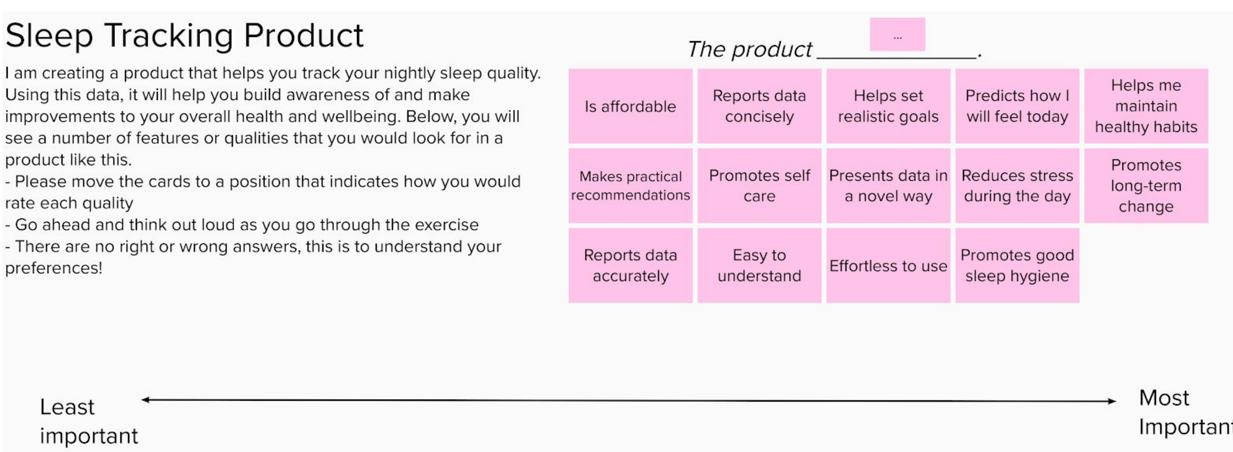


Figure 8: Interview framework for ranking user needs

Each interview resulted in a survey output that demonstrated the interviewees relative preferences as seen in figure 9. Additional comments and questions were recorded and used to refine the wording of the user need statements. In some cases, the user comments resulted in consolidating user needs statements because of overlap.

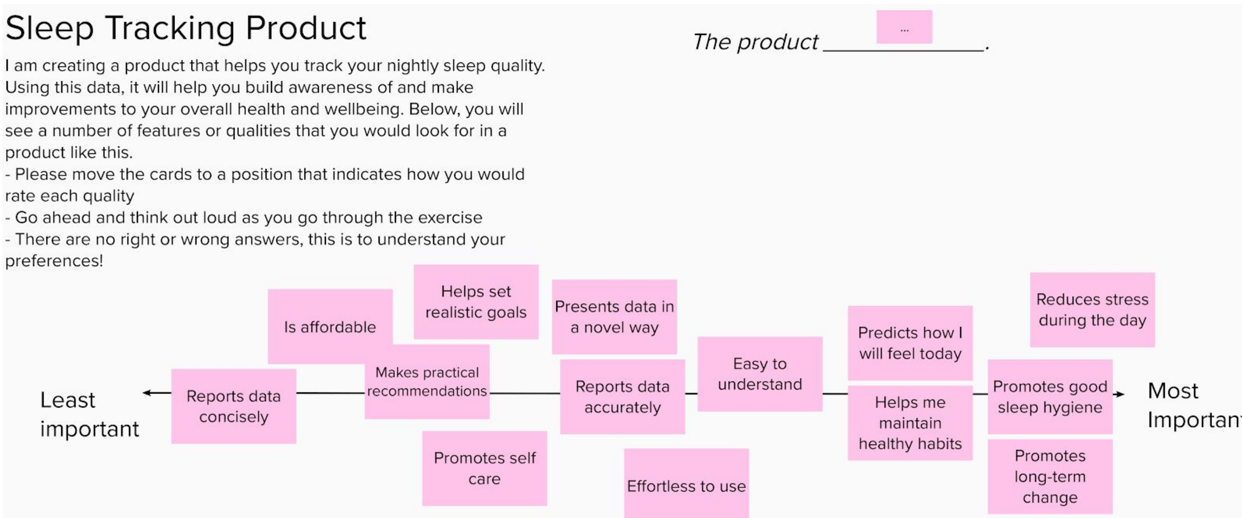


Figure 9: Example of completed user needs exercise.

Because relative weighting is used to evaluate ideas later in the downselection phase, the user needs are organized into three levels of importance based on the output of the ranking exercises.

Level 3 - Somewhat important	Level 2 - More important	Level 1 - Most important
• Able to share with loved ones • Is affordable • Makes practical recommendations	• Focuses on long-term change • Easy to understand • Helps follow realistic goals • Promotes sleep hygiene	• Nudges healthy habits • Predicts how well my day will go • Focuses on long-term change • Reduces stress during the day/promotes self care • Integrates well with my daily routines

Table 2: Prioritized user needs

Chapter 3: Concept Development and Selection

3.1 Approach

To arrive at the best possible concept, the ideation phase is broken into four stages:

a. concept generation: the researcher brainstorms possible solutions to address user needs
b. concept screening: a screening methodology is employed to screen ideas that did not address user needs
c. concept selection: a final round of stakeholder feedback is gathered to select to the final concept
d. design considerations: a loose definition is created of the product architecture of the final concept

3.2 Concept generation

A sketching session was used to brainstorm as many ideas as possible to create a consumer wearable interface that addressed latent and explicit wearable user needs. The highest prioritized user needs along with highlighted quotes were used as brainstorm prompts: "I am a wearable user and I want an interface to nudge healthy habits, predict how well my day will go, focus on long-term change, reduce stress during the day, and integrate well with my daily routines." The researcher created sketches on post-it notes to produce as many 15 unique concepts, each including a made up product name and headline to keep track of the content of each concept.

Figure 10: Brainstorming concept sketches

Reflecting on the brainstorming process, a potential flaw in this approach is the solitary nature of the concept generation and sketching period, as research has shown that methodical team-based concept generation exercises increase the quantity, quality, novelty, and variety of concepts (Linsey et al, 2011). To address potential shortcomings, additional stakeholder feedback would be engaged later in the selection process to address any potential shortcoming in the brainstorming process.

3.3 Concept screening

A Pugh screening matrix (Pugh, 1990) was used to select a few of the 15 initial concepts. The user needs defined in Chapter 2 were employed as screening criteria, as seen in Table 3. Each interface concept was rated as “better than” (+1 points) “same as” (0 points) or “worse than” (-1 points) current existing interfaces. Though this criteria is not precise, Pugh and subsequent researchers argue that this stage of the process doesn't require more precise concept scoring because "each concept is only a general notion of the ultimate product, and more detailed ratings are largely meaningless" (Ulrich et al, 2019). A slight modification to the Pugh screening process was made to give weight to the highest priority as seen in the previous chapter in table 2. Typically, weighting isn't used until later in the process; however, sufficient research was collected to understand the relative importance of the criteria to be incorporated at this stage of the design process..

	Concepts:	Daily Self Care Package	Digital Coin Jar	Fractal art installation	Health Terrarium	Healthy Houseplant	Lifespan Calculator	Morning Light Therapy	Mountain Climber
Description	Weight	Get a custom self care routine based on last night's sleep and other health indicators.	Make bets and win/lose against yourself or your friends based on activities and sleep. Lost money goes to healthcare charities.	Reveal beautiful 3D sculptural art. Materials are formed/manipulated from your health data.	Controls the plant environment based on your activity and sleep habits. Better habits grow your plants. Bad habits kill your plants.	Robotic avatar changes it's bloom and stem position to reflect your sleep and overall wellness.	A scale shows the long term impact of your daily or weekly habits if you were to continue your current state.	Wake up to different auras of light based on how well you slept.	Visual display of someone climbing a mountain. Get to the top by completing daily activities such as steps.
Nudges healthy habits	3	1	1	-1	0	0	1	-1	1
Predicts how well my day will go	3	1	0	0	-1	1	-1	1	0
Reduces stress during the day/Promotes Self Care	3	1	-1	1	1	1	-1	1	-1
Integrates well with my daily routines	3	1	0	0	1	1	-1	1	0
Focuses on long-term change	2	-1	0	0	1	1	1	-1	-1
Easy to understand	2	1	1	-1	-1	0	-1	-1	0
Helps follow realistic goals	2	1	1	-1	0	-1	-1	-1	1
Promotes sleep hygiene	2	0	0	-1	0	0	0	0	1
Able to share with loved ones	1	-1	1	1	1	1	-1	0	-1
Is affordable	1	1	1	-1	0	1	1	0	1
Makes practical recommendations	1	1	0	-1	-1	0	1	-1	1
Score		6	4	-4	1	5	-2	-2	2
Weighted Score		15	6	-7	3	11	-7	-1	3
Continue?		Yes	No	No	No	Yes	No	No	No

	Concepts:	Personal Battery	Pin display	Sleepagatchi	Sleepy Bear	Today's 5K, 10, Marathon	Wellmoji	Wellness Puzzle	Wellness Visualization Mirror Display
Description	Weight	Physical/visual battery display that charges up with sleep and other habits	Your health data is worn on you and changes based on your health data. Shows others how you are feeling.	Keep your cute avatar healthy with better sleep and self care.	Cute and furry customizable bear avatar demonstrates the sleep and exercise habits you SHOULD be taking.	Determine what you need to accomplish and watch your avatar go on the race. Your progress is connected to your health metrics.	Displays an avatar on your phone or in your home that reflects your sleep and exercise.	Can you put together the puzzle. Each piece is connected to a metric and reveals a beautiful image as you reach goals.	AR data visualization of your sleep and activity data and goals when you look in the mirror.
Nudges healthy habits	3	0	-1	1	1	1	-1	1	1
Predicts how well my day will go	3	1	-1	0	0	0	1	-1	1
Reduces stress during the day/Promotes Self Care	3	0	0	0	0	0	0	0	0
Integrates well with my daily routines	3	1	0	0	0	0	-1	-1	1
Focuses on long-term change	2	-1	-1	1	1	0	-1	0	0
Easy to understand	2	1	-1	0	0	0	1	-1	1
Helps follow realistic goals	2	0	-1	0	0	1	0	1	1
Promotes sleep hygiene	2	0	-1	1	1	1	0	0	0
Able to share with loved ones	1	1	1	-1	1	-1	1	-1	0
Is affordable	1	1	-1	1	0	1	1	0	-1
Makes practical recommendations	1	-1	-1	1	1	1	0	1	1
Score		3	-7	4	5	4	1	-1	5
Weighted Score		0	0	0	0	0	0	0	0
Continue?		No	No	Combine		No	No	No	Yes

Table 3: Pugh screening matrix

The scores were totaled, which resulted in the following five concepts:

a. Healthy House Plant: Robotic avatar blooms and changes its stem position to reflect your sleep and overall wellness.
b. Wellness Visualization Mirror Display: AR data visualization of your sleep and activity data and goals when you look in the mirror.
c. Sleepagatchi: Keep your cute avatar healthy with better sleep and self care.
d. Sleepy Bear: Cute and furry customizable bear avatar demonstrates the sleep and exercise habits you should be implementing.
e. Daily self care package: Get a custom self care routine based on last night's sleep and other health indicators.

Reflecting on the concepts that remain, the Sleepagatchi and Sleepy Bear concepts were combined because of concept overlap—both employ a creature as an avatar to display data and motivate a user to make healthier choices. Despite its high score, a “Daily self care package” concept is actually a service, not an interface. Therefore this concept was disqualified for this project, but was noted as an opportunity for further exploration.

3.4 Concept selection

Another round of stakeholder feedback was gathered to select the final concept. The sleep tracking use case was selected due to its popularity in the initial round of interviews. Choosing one use case helps flatten differences between the interface concepts and eliminates some feedback bias—each interface was redrawn to demonstrate two figurative states: 1) a rested state and 2) an unrested state. Research by Macomber and Yang (2012) shows that stakeholder feedback varies depending on the style and level of finish of hand sketches of design concepts, especially when comparing “rough” and “clean” hand sketches. The original sketches varied in style, so the final concept choices were redrawn to give the same style of appearance before they were shown to interview subjects to reduce bias in feedback.

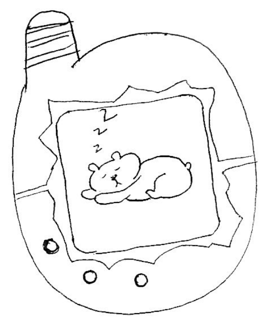
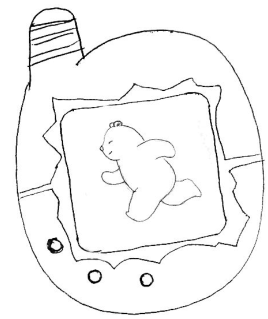
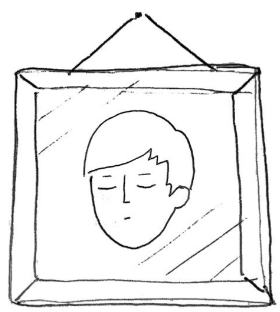
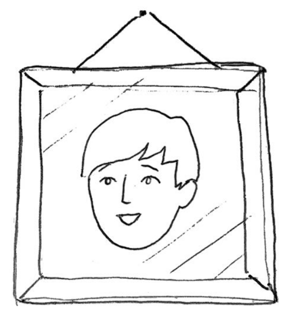
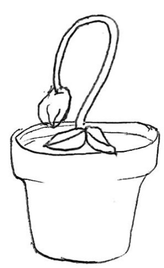
	Unrested State	Rested State
Concept: Sleep Avatar *A hand-held digital avatar companion shows you how well you slept last night*		
Concept: Visualization Mirror *An augmented mirror shows you how well you slept last night*		
Concept: Robotic House Plant *Robotic household plant reflects how well you slept last night*		

Figure 11: Concept Sketches for User Feedback

Three of the seven interviewees were available for a five-minute video call to review the concepts and share feedback. The interview protocol went as follows:

- The concepts were shown as in Figure 11 in random order on one page.

- Each concept was described briefly: “When you wake up in the morning, this is what the product looks like. The version on the right is what you see when you are well rested and the version on the left is what you see when you are not well rested.”
- Which concept best represents you when you feel rested? Which concept best represents you when you do not feel well rested?
- How would you rank these concepts for a product interface that helps you improve your sleeping habits?

Two of the three respondents selected the houseplant as the best match for both figurative representations of rest and lack of rest. Two of the three respondents selected the house plant as the most desirable product. The healthy house plant was selected as the winning concept.

3.5 Further design research

The research so far assumed an intuitive link between a flowering plant’s visual appearance as a metaphor for sleep, which could be explored much further. An additional round of interviews was conducted to better understand how consumer wearable users respond to visual attributes of plants and how those attributes lend themselves to a metaphor of a rested or unrested person.

Five participants were selected who had previously responded to the screening survey, but did not participate in interviews. Nine stock images were selected to represent varying flower species (colors, petal shapes, sizes), with differing stem positions (rigid stem, bent stem, or no stem visible) and types of petals (open and closed petals of different shapes and sizes). Interviewee responses were recorded for each stimuli. This qualitative approach quickly captured the range of characteristics that participants notice when encountering a flowering plant and ascribing its wellbeing.

Interview Script:

- Researcher Introduction: "I'm going to show you nine images of different flowers and would like you to respond to each one to tell me how healthy they look. There are no right or wrong answers, and this is not a quiz on plant taxonomy. Your response can be as short or as long as you would like. At the end, we are going to review the flowers and I'm going to ask you a couple of additional questions."

- Section 1 protocol: Show each flower image separately. Provide ample time for the interviewee to respond to each flower. Ask "why?" to encourage the participant to refer to elements of the flower when providing his or her reaction to the flower.

- Section 2 protocol: Each flower image is shown on one slide. The images are numbered for easy reference during conversation.
    - "You wake up in the morning. You feel amazing from having a great sleep and you are ready to take on the day. Which flower represents how you feel? Why?"
    - "You wake up in the morning. You don't feel well rested because you didn't get good sleep. Which flower represents how you feel?"

The responses were collected and assigned to each flower to better understand the construct of restfulness and how it is assigned to different flower images as seen in tables 5 and 6.

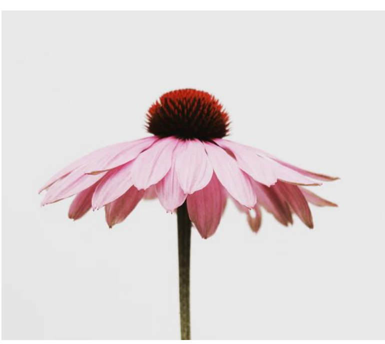
Plants Selected as Figurative Examples of Good Sleep		
Stimuli	*Section 1 Responses*	*Section 2 Responses*
 (Raclette, 2017)	"This is a wild, adventury, and maybe even hippy kind of vibe." "You don't cultivate [this flower], but stumble upon it." "Nature took its course. They haven't been cultivated by someone to be beautiful like the others. " "It might get lost and I wouldn't notice it if I was walking down the street." "It reminds me of freshness, but I don't relate to it because I haven't been in a field like that before." "I identify with this. Like being in a field. Freedom, wilderness and nature. However, poppies have a certain connotation." "It makes me think of freedom. It's in such a big area, not packed in. It looks young, but fragile."	"This one represents me [when I am well rested]. You can see it in the context. It's not a lonesome thing. It's in the real world." "The surroundings make me feel like I'm ready to start my day."
 (Siepel, 2017)	"I feel like the petals are going to fall off. Like they could fall off at any second." "Feels fun for some reason. Reminds me of a dandelion where you could blow on it and it would dissipate. It's fun." "I like that I can see the individual petals. It feels positive and happy. The vibrant red is very beautiful." "It's very open. There is so much air to get so it opened itself."	"I'm energized and ready to go."

Table 5: Example responses to figurative examples of good sleep

<table>
<tr><th colspan="3">Plants Selected as Figurative Examples of Bad Sleep</th></tr>
<tr><td>Stimuli</td><td>Section 1 Responses</td><td>Section 2 Responses</td></tr>
<tr><td>
(Boss, 2019)</td><td>“This one is depressing. It’s supposed to be a sunflower so you feel like it should be standing up.”

It needs some love and attention.”

“This one has had a tough life. It’s a sunflower. I want it to be happy and shining, but it has dark times ahead. Impending doom.”

“This one is a little sad.”

“It’s waiting for the sun. It will be okay, but now it is gloomy. Still standing.”</td><td>“Feels like my lack of sleep is a result of my own action. It looks like it bloomed then it fell apart. I was standing up the day before, but then I did something stupid.”</td></tr>
<tr><td>
(UJI, 2018)</td><td>“Unlike other down ones, it feels like it’s supposed to be that way.”

“It looks more purposeful. It’s down, but not wilted.”

“But it gives me the sense of melancholy. A cold dark place, but this flower is shining through.”

“This makes me feel like I’m out there in the wild. This is a tiny little flower in the midst of shrubs. It makes me feel like I've taken a walk in the rainforest.”</td><td>“I still have the potential to be beautiful, but need a little bit of ‘oomph.’ I need to accept that that’s what I’m going to be today.”

“I think this one communicates melancholy very well.”</td></tr>
</table>

Table 6: Example responses to figurative examples of good sleep

Statements that include descriptions of the plants were recorded and organized into themes. Each theme was then explored through brainstorming to consider how the theme could be expressed through mechanisms or end effectors (labeled “design opportunities” in Table 7). Then the feasibility of each idea was also evaluated based on the following two factors:

1. Ability to implement the concept to actuate robot between two states (rested state and unrested state).

2. Ability to work with materials with minimal lab space or special equipment (at the time designing this robot, all makerspaces at MIT were closed to the public due to COVID-19).

Theme	Descriptions/States	Design Opportunities	Feasibility
Detail	Simple, Ornate, Complicated	Origami	Medium
Beauty	Pretty, Ugly	Materials, Animation	Low
Petal aperture	Bloomed, Wilted, Down, Closed, Open	Servo actuation, Origami	High
Plant context	Home, In the wild	Stand, base or pot Surrounding plants/grass	High
Stem rigidity	No stem, curved down stem, straight stem	Spine with servo actuation, flexure	High
Color	Faint colors, vibrant	LED Light, Materials selection	Medium
Age	Young, fresh, old	Materials selection	Low
Strength	Fragile, growth	Materials selection	Medium

Table 7: Flower design opportunities

Upon review of the various design opportunities, the petal aperture and stem rigidity were the most feasible application for the design and would be explored further in the prototyping context. Plant context can also be a design consideration for the housing of the product, but may not include any mechanism or end effectors. Other themes will be considered in the design, but will not contribute to the primary mechanism of changing states between the rested and unrested states of the flower.

Chapter 4: Prototyping and Testing

4.1 Approach

To prototype the design concept of a robotic flower as a figurative interface to present sleep data, the concept is broken into three parts: 1) a flower that blooms, 2) a stem that changes posture and 3) a networked control system. The control system may be the most critical module of the design because it is unknown if the interface could actually be networked to retrieve data from the consumer wearable. This prioritized the fabrication of a works-like prototype of the control to understand if necessary data could be collected for the concept to work.

The petal and stem design are prototyped second, using a less technical approach. Paper prototyping is used to create approximations of functionality to gain initial feedback from users. In a study of 18 expert designers given a brief period to generate concepts for a design challenge, physical prototypes are found to outperform other design methods such as CAD modeling and sketches when shown to stakeholders for feedback (Häggman et al, 2015). These prototypes are used to better understand how the design could produce motivational affordances that would encourage behavior change.

4.2 Electronics design and production

Looking at several activity trackers, there are two opportunities to network a third-party device: as a client for the devices Bluetooth Low Energy (BLE) protocol or connecting to a cloud server via API. To explore these opportunities, the Espressif ESP32 WROOM microcontroller is selected because it has both BLE and Wifi networking capabilities according to the datasheet produced by Espressif (2019). More information about the PCB design and fabrication can be found in Appendix B.

4.2.1 Connecting interface to wearable via BLE

Now the board is fabricated and tested. BLE is an intuitive solution because activity tracking wearable devices all use BLE to broadcast data to user phones, which is then uploaded to a cloud-based server as shown in chapter 1.2. Programming the ESP32 to be a BLE client is a simple way to constantly collect data from the wearable as long as there is proximity between the two devices.. BLE is also very energy efficient and less likely to experience interference from competing signals than traditional Bluetooth.

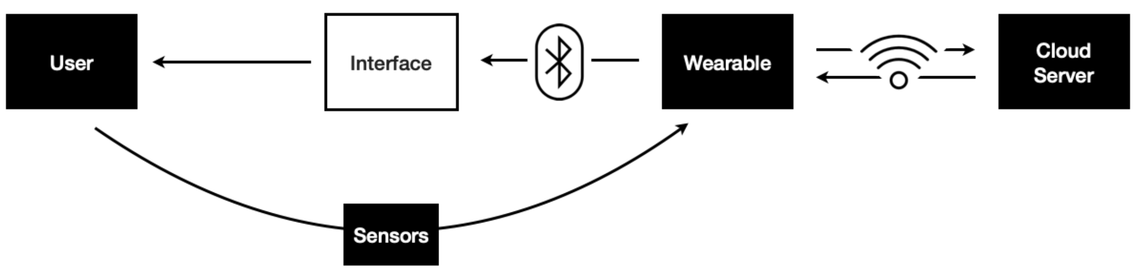


Figure 12: Block Diagram of BLE Networking Connection

The ESP32 is programmed using Arduino IDE because there were pre-existing libraries and examples that allowed data collection from a BLE host. Connecting and interpreting meaningful information from a wearable device requires customization for different devices. Several BLE scanner apps are available that find ambient signals and can determine the device's unique identifier, which is required to access and display relevant data from the wearable (See Appendix C for Arduino code). To test functionality, the researcher runs in place, which demonstrates that the ESP32 is collecting heart rate data sent from a wearable device. There is a slight delay—which isn't an issue for the final concepts for this project as they don't require instantaneous feedback.

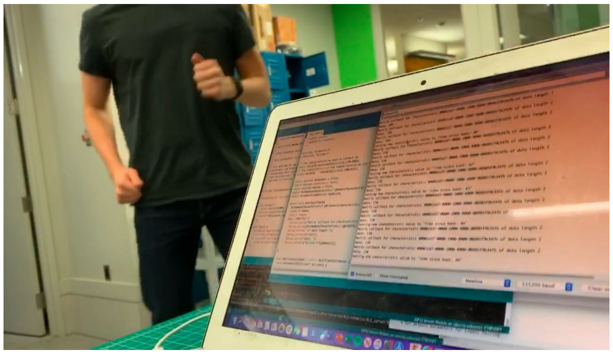


Figure 13: Testing BLE networking connection to wearable device

A simple LED array interface is fabricated to further test the networking concept. The light blinks with different frequency as heart rate increases or decreases.

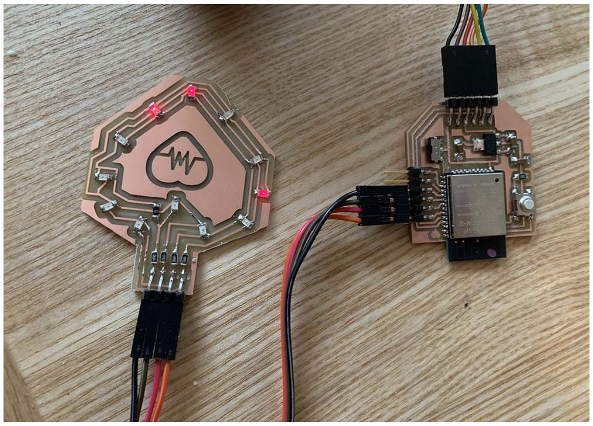


Figure 14: Prototype visual display for wearable BLE networking

Third-party BLE connection lacks a lot of the wearables' valuable data, which poses problems for this project. Though wearables send all data to the user's phone using BLE, this data is not available to be easily read by other devices. Heart rate broadcasting is a unique use case because many stationary bicycles, treadmills and even monitors at fitness boutiques have the capability to be a BLE client and display the user's heart rate. For this reason, wearable companies are willing to broadcast this data with the consent of the user. All that is broadcasted is the heart rate, and no other data related to the signal from the sensors on the device.

4.2.2 Connecting interface to wearable via API.

Most of the signal processing from sensors in the wearables used is actually processed in the cloud and not natively in the device or smartphone (see figure 15). Many wearable activity trackers allow third parties to access user data through application program interfaces (APIs). However, the availability, documentation and governance of these APIs vary significantly across products. The Fitbit is chosen because it appears, at the time of writing this, to be among the most mature and well documented APIs, dating back to 2011. An application is registered to dev.fitbit.com/ in order to secure an authorization token to access data from a test account for the Fitbit.

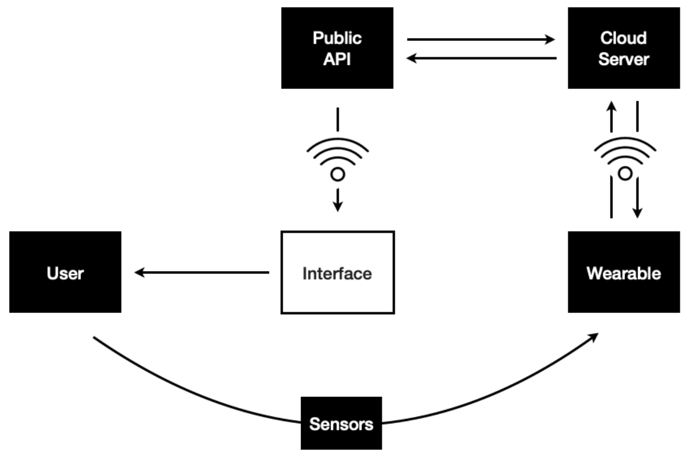


Figure 15: Block diagram of Wifi/API networking connection

Fitbit tracks sleep data in two ways according to publicly provided API documentation (2020):

- Sleep stages: Levels data is returned with 30-second granularity. 'Sleep Stages' levels include deep, light, rem, and wake.
- Sleep pattern: Levels data returned with 60-second granularity. 'Sleep Pattern' levels include asleep, restless, and awake.

This data will offer more than is needed for the proof of concept as seen in the code below, which includes date of sleep, sleep duration, restlessness, minutes awake, and sleep stages (for premium fitbit users). The API is successfully tested using the researcher's sleep data (see Appendix D for example code):


```
{"sleep":[{"awakeCount":1,"awakeDuration":4,"awakening
sCount":5,"dateOfSleep":"2020-05-01","duration":277800
00,"efficiency":98,"endTime":"2020-05-01T09:02:30.000"
,"isMainSleep":true,"logId":26978710301,,"minutesAfter
```

Wakeup":0,"minutesAsleep":455,"minutesAwake":8,"minute
sToFallAsleep":0,"restlessCount":4,"restlessDuration":
4,"startTime":"2020-05-01T01:19:30.000","timeInBed":46
3}],"summary":{"stages":{"deep":0,"light":0,"rem":0,"w
ake":0},"totalMinutesAsleep":455,"totalSleepRecords":1
,"totalTimeInBed":463}}

This amount and granularity of data is more than enough for the purposes of this project. In fact, testing the interface will just involve using the number of hours of sleep as a simple and meaningful proxy of sleep quality. However, in the future it would be important to work with a sleep scientist to engage peer-reviewed research to further account for critical factors such as sleep latency, number of awakenings, wake after sleep onset, and sleep efficiency (Ohayon et al, 2017).

4.2.3 Networking design selection

Upon looking at both networking options, connecting to the wearable API using Wifi will be a much more robust data collection method, which will be used for this project. Some challenges with the API remain, however. The documentation of the APIs vary greatly across different consumer wearables, which may make it difficult for this interface to be interoperable across different device types. From a tactical standpoint, the authentication token for the API is often revoked. In the future, to increase the robustness of the connection, developing a standalone API that connected to the Fitbit API would be a more durable solution.

4.3 Flower design targets

In chapter 4.5, mechanical elements were selected that could most effectively and feasibly demonstrate rest: 1) flower's petal aperture and 2) stem rigidity. A third construct of plant context will be considered later in the assembly chapter.

To better understand the target states of a well rested vs. an unrested flower, various angles of petal aperture were reviewed to understand how these postures communicated plant restedness. As seen in the photos from the plant study, petal aperture ranged from beyond 180° all the way to zero. A graphical representation is created to visualize the various angles and identify the appropriate range.

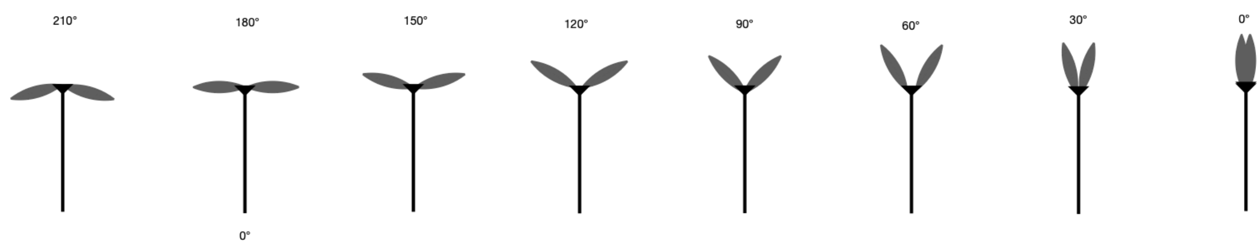


Figure 16: Petal aperture (blooming) options

Further, the tilt of the stem is also considered in eight positions, ranging from no curve to 210° stem curve. Here, stem curvature is calculated by the angle that the face of the flower is rotated from its original upright position.

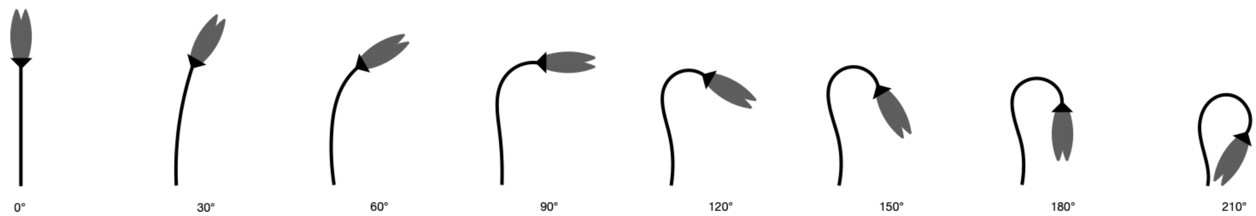


Figure 17: Stem curve options

Combining all possible petal apertures and stem curves, 64 unique permutations are considered to identify the rested- and unrested-looking extremes. As seen in figure 18, the target posture of the rested flower is 210° petal aperture and 30° stem curve. The target unrested posture of the rested flower is 0° petal aperture and 180° stem curve.

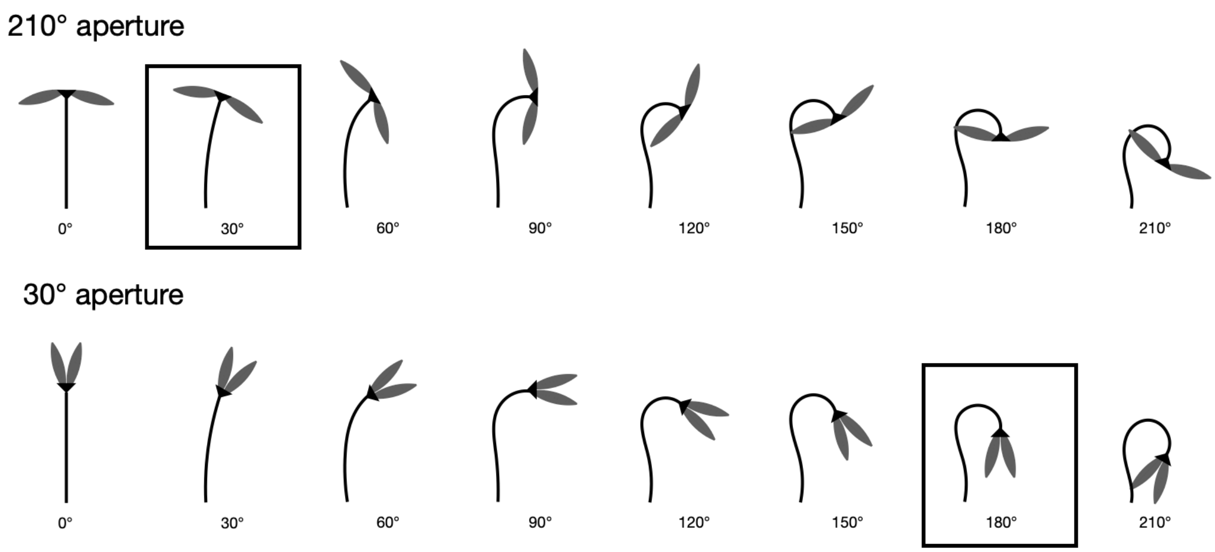


Figure 18: Selected petal apertures and stem curves

The researcher selected these postures based on personal opinion to approximate the target design. Conjoint survey analysis should be conducted to understand how a representative sample of potential interface users understand which flower postures (and other plant attributes) represent the most rested and unrested extremes..

4.4 Petal mechanism design

The design target for the petal mechanism is to control a mechanical flower to move from closed (0° petal aperture) to open (210° petal aperture). Paper prototypes are created to experiment with various mechanisms for opening and closing flower petals. Some methods took inspiration from origami, while other methods used materials like glue, pipe cleaners and silicone tubing.

Figure 19: Paper prototypes of petals

The initial round of prototyping identifies some key challenges. Wider petals look appealing, but are more challenging to change positions as they crowd each other when the flower is closed. Higher numbers of petals have the same effect.

The most effective designs fall into two categories as seen in figure 20.:

1. Tether mechanism: the back of the petals are attached to a tether that opens the petal aperture when pulled back. When the tether is released, the flower closes.

2. Cone mechanism: a cone underneath the petals applies pressure that forces them to close. When the cone is lowered, the petals open.

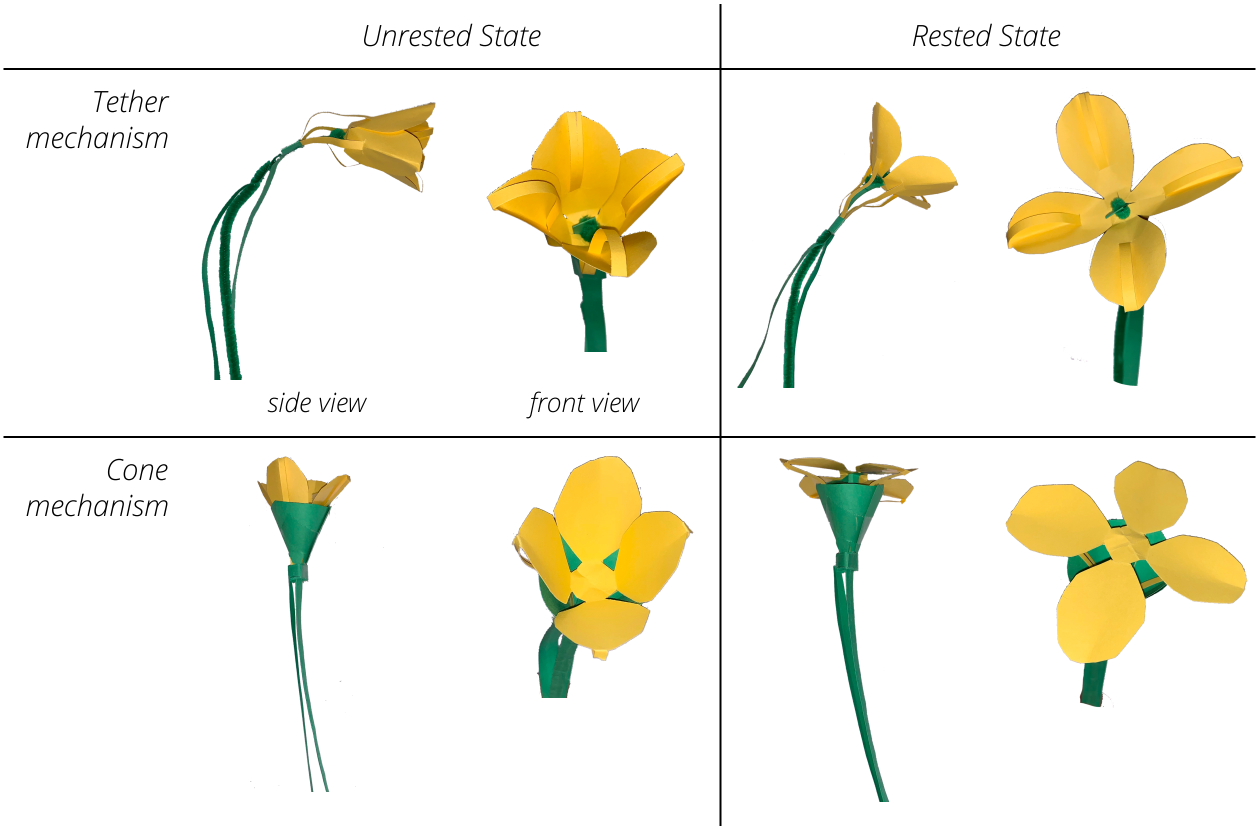


Figure 20: Visualizing various petal apertures and stem curves

The tether mechanism is best suited for the overall design because it could more easily be incorporated with the stem mechanism, which is discussed in chapter 4.5.

4.5 Stem mechanism design

The prototype stem is made by pulling a single thread through a sequence of five beads (each 9 mm in diameter and height) to change between a collapsed position and upright, rigid state as seen in figure 21.

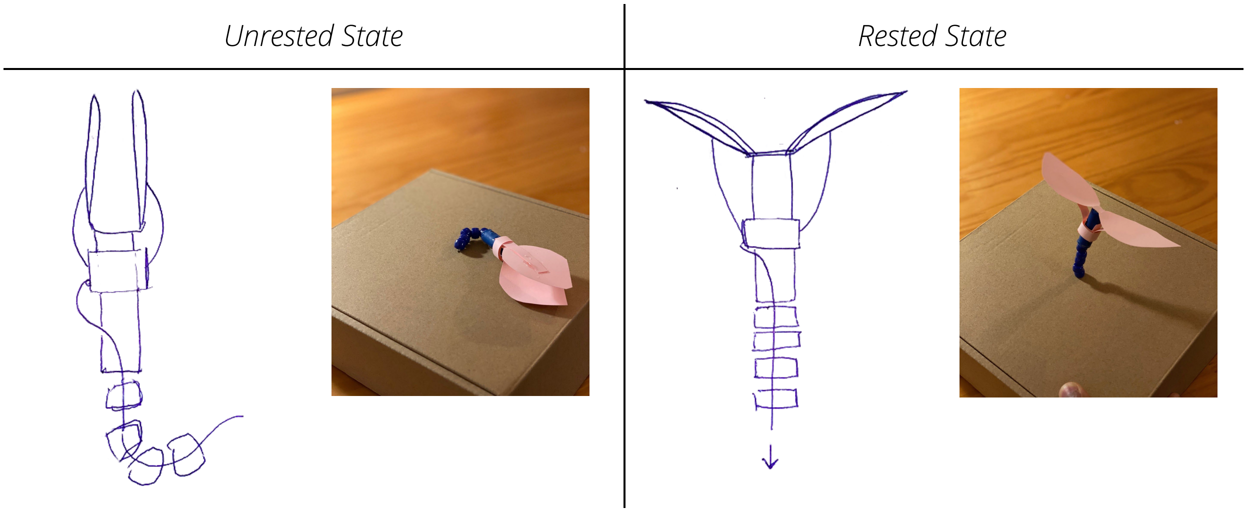


Figure 21: Stem sketches and low fidelity prototype

The flower blooming position is considered in the stem prototype to determine if the same mechanism that changes the position of the stem could also bloom the petals using the tether mechanism from chapter 4.4. In this initial test, the prototype successfully opens the flower when the string is pulled, but the flower doesn't return to a closed state when the string is released. Ensuring the flower can return to its unopened state after blooming is an important consideration for the assembly stage.

The initial prototype showed that different postures can be achieved by pulling the string, but the final design needed to be able to display a continuous range of positions depending on the amount of sleep the user had the night before. Three features were added to the beads as seen in figure 22 to ensure the bead and string design could achieve a range of poses.

- A new asymmetric bead design incorporated features similar to vertebrae in a spine, to ensure the beads remained stacked and went in one desired direction when the string loosened.
- Instead of a flat surface on the bottom, part of the surface slanted at 18°, so when the sting loosens, 10 beads would make a 180° arc.
- A second hole is added to the beads for additional stability. A string running through the back of the spine would be used to control the posture and the petal bloom. A second string running through the front of the spine would be used to stabilize the spine and ensure it went into the same position when the back string is loosened.

Twelve test beads are created from the CAD model using a fused-deposition molding (FDM) 3D printer to test if the spine design would work.

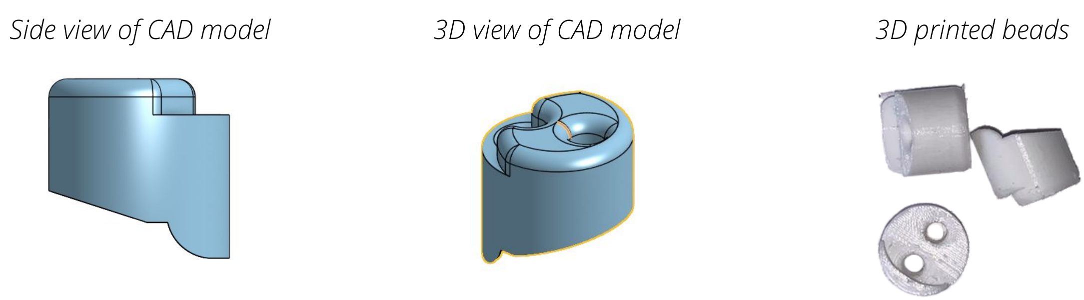


Figure 22: Stem bead CAD design and 3D print

Ten beads are tested to achieve the desired 180° arc. However, the design fails as the vertebrae in the spine are unstable and dislocate when the string is pulled. The spine successfully changes postures without dislocation with eight beads, as seen in figure 23, suggesting the geometry of the beads will work if slightly modified to increase stability.

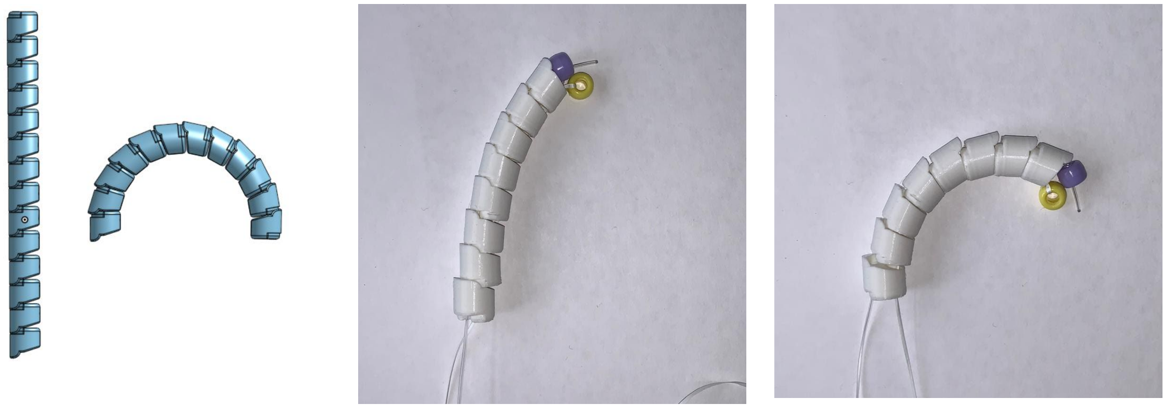


Figure 23: Customized stem bead prototype testing

Finally, the paper flower prototype is tested together with the stem beads as seen in figure 24.

Unrested State | *Rested State*

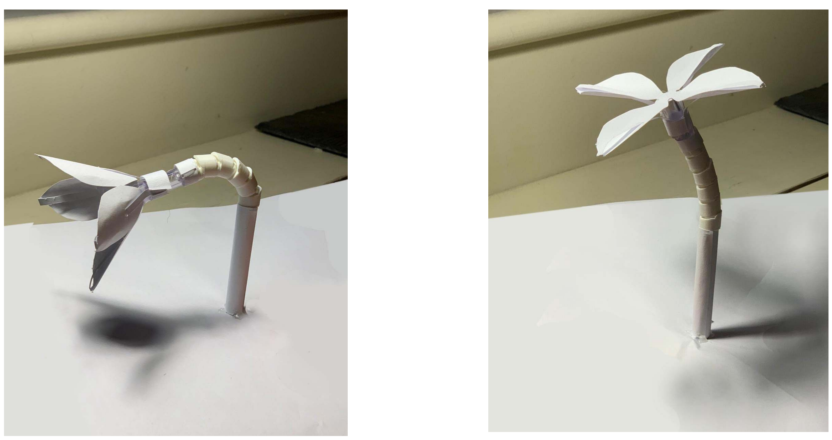


Figure 24: Assembled prototype

The target postures are not completely achieved with the current design. The petal aperture does not close all the way to 0°, and the stem lacks the number of beads to reach the 180° curve. Additional iterations of these mechanisms will achieve the control that is required and reduce the number of failure states found with the current design.

4.6 Control system

At the base of the stem, the string that runs through the back of the spine is attached to a servo. Rotating the servo will pull on the string and that tension will straighten the spine and pull on the petal tethers to bloom the flower. For this prototype, a crude metric of sleep quality is used by changing actuation based on hours of sleep. Sleep quality is divided into five states, ranging from worst (shortest sleep duration) to best (longest sleep duration).

The National Sleep Foundation recommends 7-9 hours of sleep for adults ages 26 to 64 (NSF, 2015). This is used as a guidepost, as eight hours or more is determined to be the highest quality category of sleep. Each category below is one hour less sleep, and the lowest category is four hours or less. Each category is assigned a servo rotation to bloom the flower and change stem positions that reflects the quality of sleep.

State	1 (worst)	2	3	4	5 (best)
Sleep duration	hours < 4	$4 \leq hrs < 5$	$5 \leq hrs < 6$	$6 \leq hrs < 7$	$hrs > 480$
Servo rotation	36°	72°	108°	144°	180°

Table 8: Sleep quality duration spectrum

These five categories are programmed to control a nine gram micro servo in Arduino IDE. This code is an excerpt from the firmware code found in Appendix D.

```
 if(time_sleep > 480)
{
  fitbit_servo.write(180);
  Serial.println("180 degrees");
}
else if(480 > time_sleep >= 420)
{
  fitbit_servo.write(144);
  Serial.println("144 degrees");
```

```
}
else if(360 > time_sleep >= 300)
{
  fitbit_servo.write(108);
  Serial.println("108 degrees");
}
else if(300 > time_sleep >= 240)
{
  fitbit_servo.write(72);
  Serial.println("72 degrees");
}
else if(time_sleep < 240 )
{
  fitbit_servo.write(36);
  Serial.println("36 degrees");
}
```

The result is a crude prototype that actuates the flower posture based on duration of sleep. In the future, new metrics such as restlessness, minutes awake, and sleep stages could be incorporated in the flower posture. Further, instead of breaking sleep into five discrete levels of quality, calculating a numerical sleep quality score could be correlated with servo rotation to generate a continuous range of flower postures.

4.7 Summary of results

Following the design innovation process, we created a proof of concept of a figurative interface that displays sleep duration data for a consumer activity tracker user. The product is designed with the intention to be emotionally appealing, share feedback data in an intuitive way, integrate with the user’s daily habits and provoke long-term behavior change. In figure 24, CAD modeling is used to assemble the components and demonstrate the visual appeal of the robotic interface.

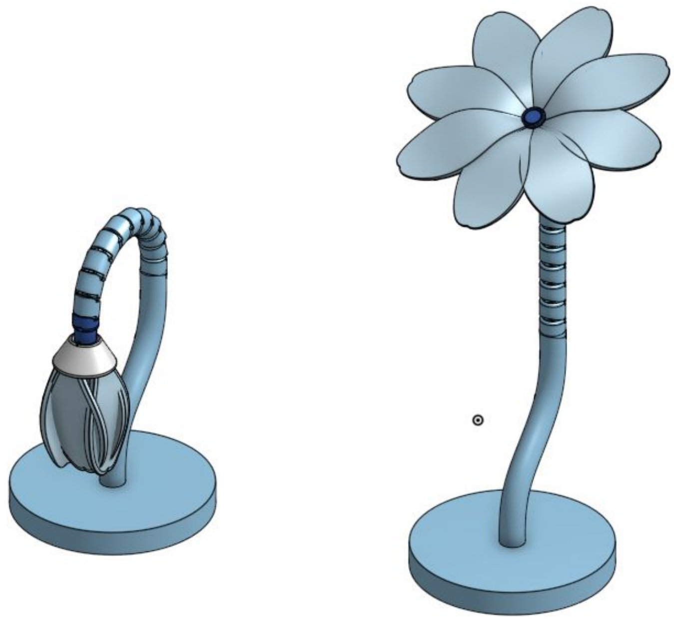


Figure 23: CAD model of final assembly

Chapter 5: Conclusion and Future Work

5.1 Future product iterations

Though each component has been tested separately (control system, stem, flower petals) to create a proof of concept, additional assembly is required to create a full working prototype. Initial testing shows that the stem and petal structure will need additional design iterations to ensure stability and control. Informal feedback to the prototype suggests opportunities to improve the visual appearance of the stem and flower. Research on the phenomenon of the "uncanny valley" shows that when robots look almost—but not quite realistically—human, it leaves an unpleasant impression on the human user (Mori, 1970; Seyama & Nagayama, 2007). Reactions to the flower robot suggest a similar response is true for impressions of plant robots. Future design iterations can explore more caricature-oriented visualizations or seek better realism to increase the desire to nurture the robotic plant and therefore improve sleep habits.

A third option that was screened out in the Pugh matrix in chapter three but may show promise is to actually grow real plants that are impacted by the user's behaviors. Plan response time is a factor that may make this challenging, but Cyborg Botany is a project at the MIT Media Lab that combines natural plant functions and digital sensing and interactions and may have applications for a hyper-realistic and compelling version of this concept (Sareen et al, 2019).

5.2 Additional research

The question remains if this type of figurative user interface is better than or can augment the original analytical interface. A higher fidelity prototype of the figurative interface needs to be empirically studied as a treatment compared to the original analytical version. In this research, the following questions would need to be addressed:

1. Do users understand what the figurative user interface is telling them?
2. Do users exposed to the figurative user interface treatment change behavior more than those exclusively using the analytical interface?
3. Does the figurative interface treatment augment or change usage of the analytical interface?

Regarding the third question, one interview respondent who viewed the final concept says "if the plant is looking happy, I don't need to look at my data—I'm good. If it's looking sad, I'm definitely going to look on the app and see what's going wrong." This suggests that multiple connected interfaces could reinforce motivational affordances such as reminders and goals that lead to higher rates of behavior change.

5.3 Future Opportunities

This project opens the possibility of an entire ecosystem of interfaces, products and services that could respond to a user's activity tracking data. The robotic flower represents a lifeform that a user may enjoy and seek to nurture, but many other products could leverage the same data to motivate a user's habits. Our lives could include a variety of objects that reflect our physiological state in figurative ways, increasing self-awareness and promoting healthy habits.

References

About Fitbit, www.fitbit.com/us/about-us.

Al, Quino. “Photo by Quino Al on Unsplash.” Beautiful Free Images & Pictures, 17 Sept. 2016, unsplash.com/photos/BlMj6RYy3c0.

Aroganam, Gobinath, et al. “Review on Wearable Technology Sensors Used in Consumer Sport Applications.” Sensors, vol. 19, no. 9, 2019, p. 1983., doi:10.3390/s19091983.

Ashe, Maureen C, et al. “‘Not Just Another Walking Program’: Everyday Activity Supports You (EASY) Model—a Randomized Pilot Study for a Parallel Randomized Controlled Trial.” Pilot and Feasibility Studies, vol. 1, no. 1, 2015, doi:10.1186/2055-5784-1-4.

Bao, Qifang, et al. “Investigating User Emotional Responses to Eco-Feedback Designs.” Volume 7: 30th International Conference on Design Theory and Methodology, 2018, doi:10.1115/detc2018-86208.

Boss, Mark. “Photo by Mark Boss on Unsplash.” Beautiful Free Images & Pictures, 11 Oct. 2019, unsplash.com/photos/VaroP6PCjLk.

Camburn, Bradley A., et al. “Design Innovation: A Study of Integrated Practice.” Volume 7: 29th International Conference on Design Theory and Methodology, 2017, doi:10.1115/detc2017-68382.

Chenxu, Han. “Photo by Han Chenxu on Unsplash.” Beautiful Free Images & Pictures, 3 May 2018, unsplash.com/photos/YdAqiUkUoWA.

Chinoy, Evan D, et al. “1002 Examination of Wearable and Non-Wearable Consumer Sleep-Tracking Devices Versus Polysomnography.” Sleep, vol. 42, no. Supplement_1, 2019, doi:10.1093/sleep/zsz067.999.

Chung, Arlene E., et al. “Tweeting to Health.” Clinical Pediatrics, vol. 56, no. 1, 2016, pp. 26–32., doi:10.1177/0009922816653385.

Clausing, Don. “Total Quality Development.” 1998, doi:10.1115/1.800695.

Cotton, Victor, and Mitesh S. Patel. “Gamification Use and Design in Popular Health and Fitness Mobile Applications.” American Journal of Health Promotion, vol. 33, no. 3, 2018, pp. 448–451., doi:10.1177/0890117118790394.

“ESP32 WROOM Series.” Overview | Espressif Systems, www.espressif.com/en/products/modules/esp-wroom-32/overview.

Elizabeth, Edwards, et al. "'Gamification' for Health Behaviour Change in Smartphone Apps." Frontiers in Public Health, vol. 4, 2016, doi:10.3389/conf.fpubh.2016.01.00043.

"Espressif/Arduino-esp32." GitHub, github.com/espressif/arduino-esp32/tree/master/libraries/BLE/examples/BLE_client.

Faste, Rolf A. "Perceiving Needs." SAE Technical Paper Series, 1987, doi:10.4271/871534.

"Fitbit Development: Sleep Logs." Fitbit, dev.fitbit.com/build/reference/web-api/sleep/.

Grandner, Michael A., and Mary E. Rosenberger. "Actigraphic Sleep Tracking and Wearables: Historical Context, Scientific Applications and Guidelines, Limitations, and Considerations for Commercial Sleep Devices." Sleep and Health, 2019, pp. 147–157., doi:10.1016/b978-0-12-815373-4.00012-5.

Griffin, Abbie, and John R. Hauser. "The Voice of the Customer." Marketing Science, vol. 12, no. 1, 1993, pp. 1–27., doi:10.1287/mksc.12.1.1.

Häggman, Anders, et al. "Connections Between the Design Tool, Design Attributes, and User Preferences in Early Stage Design." Journal of Mechanical Design, vol. 137, no. 7, 2015, doi:10.1115/1.4030181.

Jakicic, John M., et al. "Effect of Wearable Technology Combined With a Lifestyle Intervention on Long-Term Weight Loss." Obstetrical & Gynecological Survey, vol. 72, no. 2, 2017, pp. 67–68., doi:10.1097/01.ogx.0000512372.67520.49.

Kahneman, Daniel. Thinking, Fast and Slow. Nota, 2013.

Kunst, Alexander. "EHealth Tracker / Smart Watch Ownership by Brand in the United States 2019 Consumer." Statista, 7 Aug. 2019, www.statista.com/forecasts/997195/ehealth-tracker-smart-watch-ownership-by-brand-in-the-us.

Linsey, J. S., et al. "An Experimental Study of Group Idea Generation Techniques: Understanding the Roles of Idea Representation and Viewing Methods." Journal of Mechanical Design, vol. 133, no. 3, 2011, doi:10.1115/1.4003498.

Macomber, Bryan, and Maria Yang. "The Role of Sketch Finish and Style in User Responses to Early Stage Design Concepts." Volume 9: 23rd International Conference on Design Theory and Methodology; 16th Design for Manufacturing and the Life Cycle Conference, 2011, doi:10.1115/detc2011-48714.

McCutcheon, Sharon. "Photo by Sharon McCutcheon on Unsplash." Beautiful Free Images & Pictures, 23 Feb. 2018, unsplash.com/photos/GqPhbmoLUIs.

Mori, Masahiro. “The Uncanny Valley.” The Monster Theory Reader, 2020, pp. 89–94., doi:10.5749/j.ctvtv937f.7.

Ohayon, Maurice, et al. “National Sleep Foundation's Sleep Quality Recommendations: First Report.” Sleep Health, vol. 3, no. 1, 2017, pp. 6–19., doi:10.1016/j.sleh.2016.11.006.

Peake, Jonathan M., et al. “A Critical Review of Consumer Wearables, Mobile Applications, and Equipment for Providing Biofeedback, Monitoring Stress, and Sleep in Physically Active Populations.” Frontiers in Physiology, vol. 9, 2018, doi:10.3389/fphys.2018.00743.

“Project Overview ' HeartBit.” MIT Media Lab, www.media.mit.edu/projects/heartbit/overview/.

Seyama, Jun'ichiro, and Ruth S. Nagayama. “The Uncanny Valley: Effect of Realism on the Impression of Artificial Human Faces.” Presence: Teleoperators and Virtual Environments, vol. 16, no. 4, 2007, pp. 337–351., doi:10.1162/pres.16.4.337.

Spratt, Annie. “Photo by Annie Spratt on Unsplash.” Beautiful Free Images & Pictures, 23 Jan. 2018, unsplash.com/photos/8dhv-lqpgK4.

“Tamagotchi.” Wikipedia, Wikimedia Foundation, 17 May 2020, en.wikipedia.org/wiki/Tamagotchi#/media/File:Tamagotchi_0124_ubt.jpeg.

Terwiesch, Christian, and Karl Ulrich. Innovation Tournaments Creating and Selecting Exceptional Opportunities. Harvard Business Review Press, 2009.

“US20170055898A1 - Determining Sleep Stages and Sleep Events Using Sensor Data.” Google Patents, Google, patents.google.com/patent/US20170055898A1/en?q=sleep+staging+heartrate+variability&oq=sleep+staging+heartrate+variability.

Uji, Tomoko. “Photo by TOMOKO UJI on Unsplash.” Beautiful Free Images & Pictures, 19 Mar. 2018, unsplash.com/photos/JoqLKJPxovk.

Ulrich, Karl T., et al. Product Design and Development. McGraw-Hill Education, 2019.

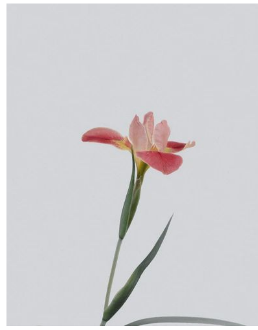
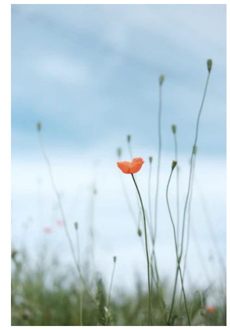
Wang, Julie B., et al. “Mobile and Wearable Device Features That Matter in Promoting Physical Activity.” Journal of Mobile Technology in Medicine, vol. 5, no. 2, 2016, pp. 2–11., doi:10.7309/jmtm.5.2.2.

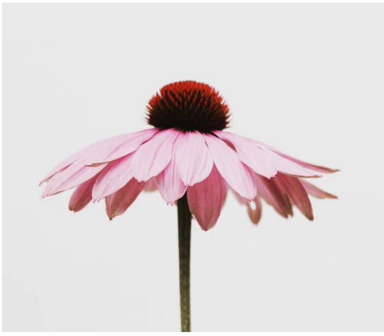
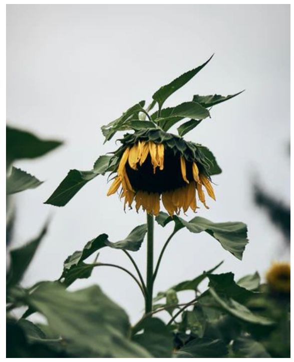
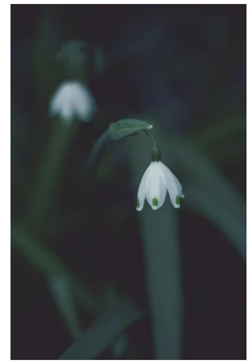
Weiser, Paul, et al. “A Taxonomy of Motivational Affordances for Meaningful Gamified and Persuasive Technologies.” Proceedings of EnviroInfo and ICT for Sustainability 2015, 2015, doi:10.2991/ict4s-env-15.2015.31.

“Fab Lab Inventory.” Google Drive, Google, docs.google.com/spreadsheets/u/0/d/1U-jcBWOJEjBT5A0N84IUubtcHKMEMtndQPLCkZCkVsU/pub?single=true&gid=0&output=html.

"Mods." Mods, mods.cba.mit.edu/.

Appendix A: Flower Study

 (Al, 2016)	 (Chenxu, 2018)	 (Raclette, 2017)
“This one is ornate. It’s complicated on the inside.” “I would imagine this is a very pretty flower when it’s on the plant. “I feel peace because it’s so still and fully bloomed.” “I like that you can see the distinct shape, it seems very intense.” “Makes me think of a Georgia O’keefe painting” “Its attractive, but doesn’t look energetic to me. Might be because of the colors.”	“It’s delicate and fresh” “It doesn’t particularly make me feel anything” “The background takes away the emotion” “I don’t like this plant because it doesn’t have strong features. The colors are kind of blah.” “Its so pure and simple. It looks young.	“This is a wild, adventury, and maybe even hippy kind of vibe.” “You don’t cultivate, but stumble upon it.” “Nature tooks it’s course. They haven’t been cultivated by someone to be beautiful like the other. ” “It’s cute. It might get lost and I wouldn’t notice it if I was walking down the street.” “It reminds me of freshness, but I don’t relate to it because I haven’t been in a field like that before.” “I identify with this. Like being in a field. Freedom, wilderness and nature. However, poppies have a certain connotation.” “It makes me think of freedom. It is such a big area, not packed in. It looks young, but fragile.”
		GOOD SLEEP: “This one represents me. You can see it in the context. It’s not a lonesome thing. It’s in the real world.” GOOD SLEEP: “The surroundings make me feel like I’m ready start my day”

 (Siepel, 2017)	 (Boss, 2019)	 (UJI, 2018)
“I feel like the petals are going to fall off. Like they could fall off at any second.” “Feels fun for some reason. Reminds me of dandelion where you could blow on it and it would dissipate. It’s fun.” “I like that I can see the individual petals. It feels positive and happy. The vibrant red is very beautiful” “It’s very open. There is so much air to get so it opened itself.”	“This one is depressing. It’s supposed to be a sunflower so you feel like it should be standing up.” It needs some love and attention.” “This one has had a tough life. It’s a sunflower. I want it to be happy and shining, but it has dark times ahead. Impending doom.” “This one is a little sad.” “It’s waiting for the sun. It will be okay, but now it is gloomy. Still standing”	“Unlike other down ones, it feels like it’s supposed to be that way.” “It looks more purposeful. It’s down, but not wilted” “That is tiny.” “This makes me feel like I’m out there in the wild. This is a tiny little flower in the midst of shrubs. It makes me feel like I've taken a walk in the rainforest.” “But it gives me the sense of melancholy. A cold dark place, but this flower is shining through.
GOOD SLEEP: “I’m energized and ready to go.”	BAD SLEEP: “Feels like my lack of sleep is a result of my own action. It looks like it bloomed then it fell apart. I was standing up the day before, but then I did something stupid.”	BAD SLEEP: “I still have the potential to be beautiful, but need a little bit of oof. I need to accept that that’s what I’m going to be today.” BAD SLEEP: I think this one communicates melancholy very well.

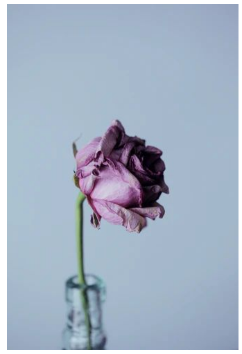
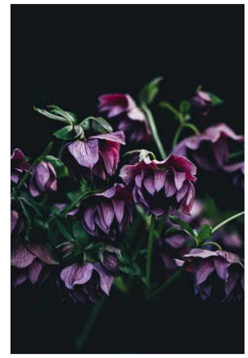
 (Spratt, 2018)	 (McCutcheon, 2018)	 (Spratt, 2017)
"Throw this shit out. Game over. Put it in the garden." "This is why I don't buy flowers." "It's been a waste. The vase is so big, but the flowers are dead. I want to clean up the space." "This is sad because, in the past, it looked really beautiful. It's not that they are old, but someone didn't take proper care of them. It's pretty depressing." "Very beautiful, but not cared for. Someone didn't care."	"This reminds me of the beauty of the beast. Reminds me of dying. "It looks like it's drying up. It looks real and like something I would keep at home. It's relatable." "It's reassuring and realistic. It is what it is." "The vision of the old dry rose makes me think of the bedroom of high school girls who got this rose for prom and you hold onto it. Positive memories that have faded."	This one is sad to me. I can't tell if that's what it's supposed to be as a flower. "It's down. It brings me down. It's not getting enough sunlight. The environment is not right." "It makes me a little sad, because it looks like it's dying. It's drooping and the green on the top of the flower looks like it's rotting." "I don't like it. It's not beautiful. The angle is not good. It doesn't look fresh, maybe because it is not looking up. I can't see inside, which is negative."
BAD SLEEP: "Feels like someone else didn't take care of it."		

Table A.1: Interview responses to nine flower stimuli

Appendix B: PCB Design and Fabrication Details

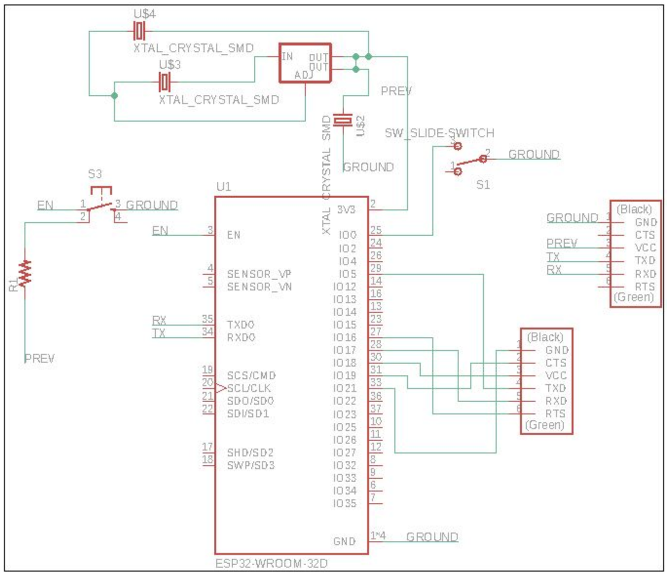


Figure B.1: PCB schematic for ESP32 controller

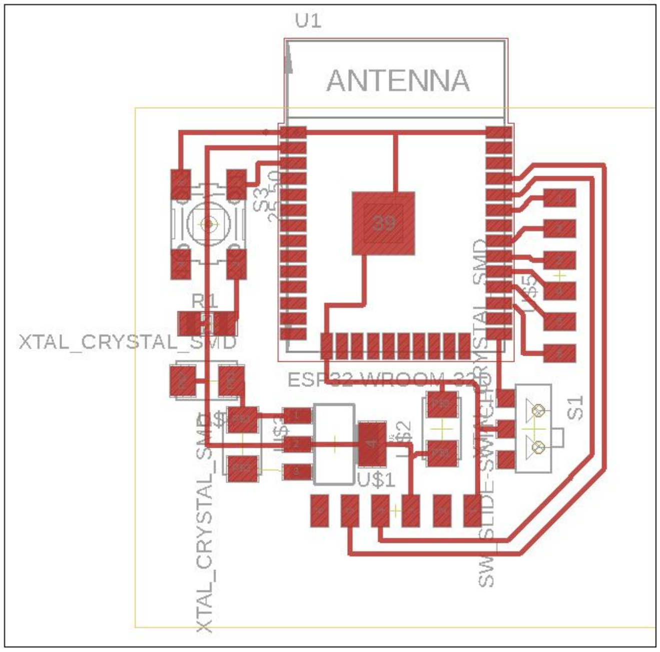


Figure B.2: PCB design diagram for ESP32 controller

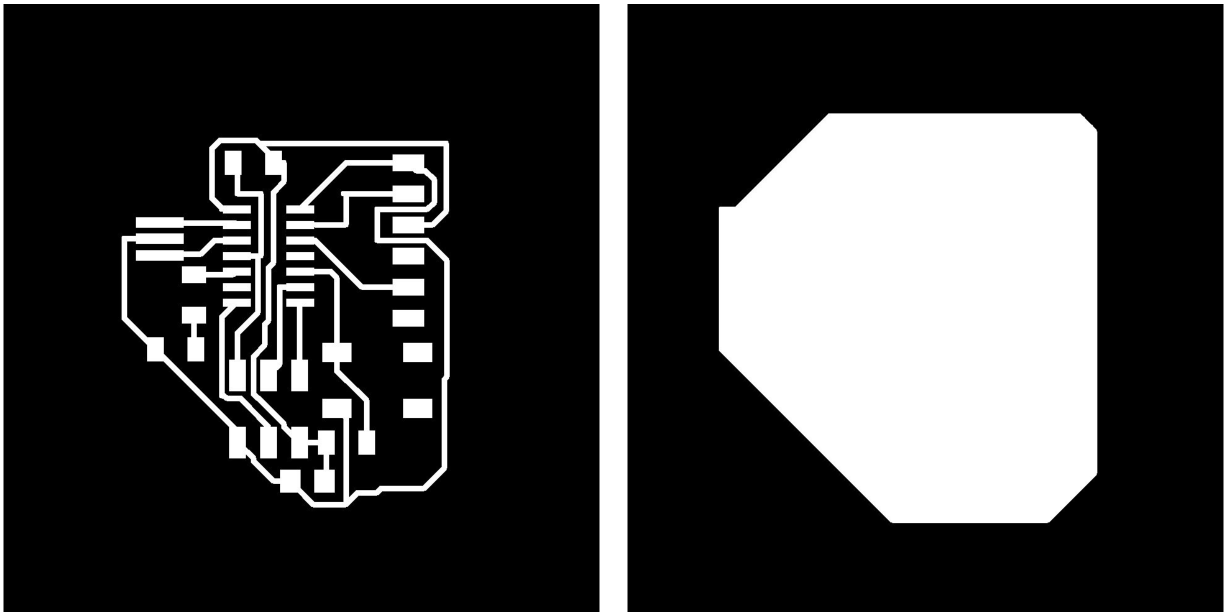


Figure B.3: PNG image used for milling machine trace (left) and outline (right) toolpaths

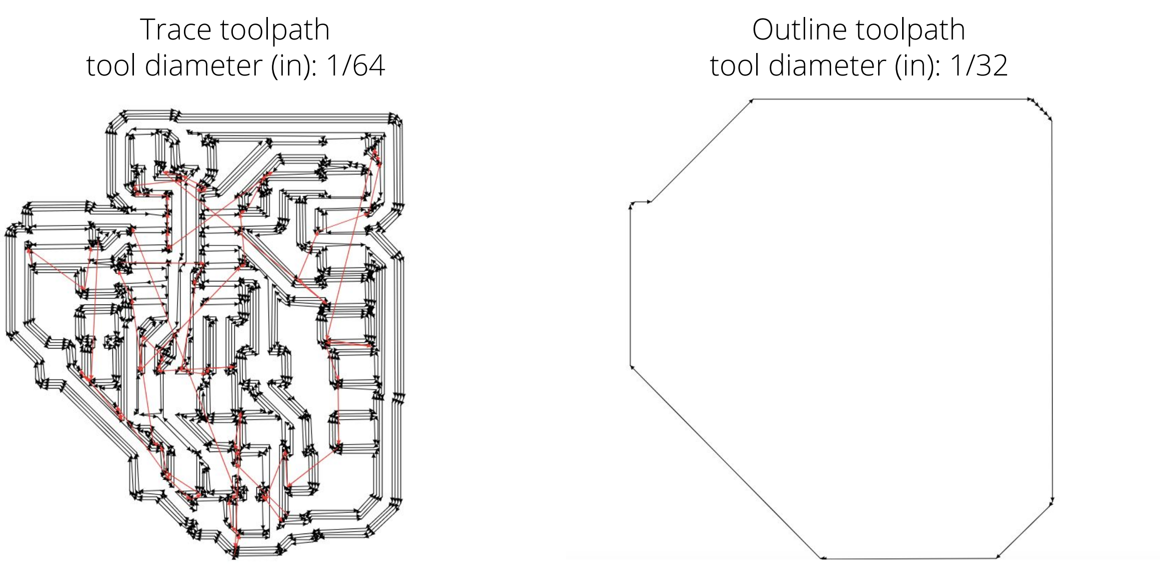


Figure B.4: PCB design diagram for ESP32 controller (MODS, 2019)

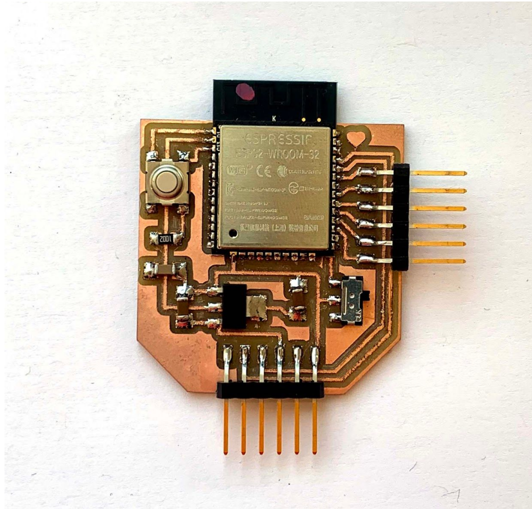


Figure B.5: Fabricated PCB boards

Component/ Material	#	Cost per unit	Total Cost
Copper PCB Board	1	$1.75	$1.75
ESP32-WROOM	1	$3.40	$3.40
Slide Switch	1	$0.84	$0.84
Tactile Switch	1	$0.74	$0.74
3.3V Regulator	1	$0.32	$0.32
Resistors	1	$0.01	$0.01
Capacitor .1 uf	1	$0.12	$0.12
Capacitor 1 uf	1	$0.07	$0.07
Capacitor 10 uf	1	$0.18	$0.18
Grand Total			**$7.43**

Table B.1: Bill of materials for components (Fablab, 2020)

Appendix C: BLE networking with LED array display code

Code adapted from Espressif BLE client example Arduino library (Github, 2020)

```
#include "BLEDevice.h"
//#include "BLEScan.h"

// The remote service we wish to connect to.
static BLEUUID serviceUUID((uint16_t)0x180D);
// The characteristic of the remote service we are interested in.
static BLEUUID    charUUID((uint16_t)0x2A37);

static boolean doConnect = false;
static boolean connected = false;
static boolean doScan = false;
static BLERemoteCharacteristic* pRemoteCharacteristic;
static BLEAdvertisedDevice* myDevice;
int heartbeat = 0;
static void notifyCallback(
  BLERemoteCharacteristic* pBLERemoteCharacteristic,
  uint8_t* pData,
  size_t length,
  bool isNotify) {
  Serial.println(*(uint8_t*)(pData+1));

}
const int LED_1 = 21;     //LED row 1
const int LED_2 = 19;     //LED row 2
const int LED_3 = 18;     //LED row 3
const int LED_4 = 5;     //LED row 4

int delaytime;

class MyClientCallback : public BLEClientCallbacks {
  void onConnect(BLEClient* pclient) {
  }

  void onDisconnect(BLEClient* pclient) {
    connected = false;
    Serial.println("onDisconnect");
  }
};

bool connectToServer() {
    Serial.print("Forming a connection to ");
    Serial.println(myDevice->getAddress().toString().c_str());

    BLEClient*  pClient  = BLEDevice::createClient();
    Serial.println(" - Created client");

    pClient->setClientCallbacks(new MyClientCallback());

    // Connect to the remove BLE Server.
    pClient->connect(myDevice);  // if you pass BLEAdvertisedDevice instead of
address, it will be recognized type of peer device address (public or private)
```

```
    Serial.println(" - Connected to server");

    // Obtain a reference to the service we are after in the remote BLE server.
    BLERemoteService* pRemoteService = pClient->getService(serviceUUID);
    if (pRemoteService == nullptr) {
      Serial.print("Failed to find our service UUID: ");
      Serial.println(serviceUUID.toString().c_str());
      pClient->disconnect();
      return false;
    }
    Serial.println(" - Found our service");


    // Obtain a reference to the characteristic in the service of the remote BLE
server.
    pRemoteCharacteristic = pRemoteService->getCharacteristic(charUUID);
    if (pRemoteCharacteristic == nullptr) {
      Serial.print("Failed to find our characteristic UUID: ");
      Serial.println(charUUID.toString().c_str());
      pClient->disconnect();
      return false;
    }
    Serial.println(" - Found our characteristic");

    // Read the value of the characteristic.
    if(pRemoteCharacteristic->canRead()) {
      std::string value = pRemoteCharacteristic->readValue();
      Serial.print("The characteristic value was: ");
      Serial.println(value.c_str());
    }

    if(pRemoteCharacteristic->canNotify())
      pRemoteCharacteristic->registerForNotify(notifyCallback);

    connected = true;
    return true;
}
/**
 * Scan for BLE servers and find the first one that advertises the service we are
looking for.
 */
class MyAdvertisedDeviceCallbacks: public BLEAdvertisedDeviceCallbacks {
 /**
   * Called for each advertising BLE server.
   */
  void onResult(BLEAdvertisedDevice advertisedDevice) {
    Serial.print("BLE Advertised Device found: ");
    Serial.println(advertisedDevice.toString().c_str());

    // We have found a device, let us now see if it contains the service we are
looking for.
    if (advertisedDevice.haveServiceUUID() &&
advertisedDevice.isAdvertisingService(serviceUUID)) {

      BLEDevice::getScan()->stop();
      myDevice = new BLEAdvertisedDevice(advertisedDevice);
      doConnect = true;
      doScan = true;

    } // Found our server
  } // onResult
```

```
}; // MyAdvertisedDeviceCallbacks

void setup() {
  Serial.begin(115200);
  Serial.println("Starting Arduino BLE Client application...");
  BLEDevice::init("");
  delaytime = 100;
  // Retrieve a Scanner and set the callback we want to use to be informed when we
  // have detected a new device.  Specify that we want active scanning and start the
  // scan to run for 5 seconds.
  // notes: scanning is happening here. trying to discover a device. whenever the
  // scanner finds a new device, it will call whatever my advertiseddevicecallback
thinks
  // it ism
  BLEScan* pBLEScan = BLEDevice::getScan();
  pBLEScan->setAdvertisedDeviceCallbacks(new MyAdvertisedDeviceCallbacks());
  pBLEScan->setInterval(1349);
  pBLEScan->setWindow(449);
  pBLEScan->setActiveScan(true);
  pBLEScan->start(5, false);
} // End of setup.


// This is the Arduino main loop function.
void loop() {

  // If the flag "doConnect" is true then we have scanned for and found the desired
  // BLE Server with which we wish to connect.  Now we connect to it.  Once we are
  // connected we set the connected flag to be true.
  if (doConnect == true) {
    if (connectToServer()) {
      Serial.println("We are now connected to the BLE Server.");
    } else {
      Serial.println("We have failed to connect to the server; there is nothin more we
will do.");
    }
    doConnect = false;
  }

  // If we are connected to a peer BLE Server, update the characteristic each time we
are reached
  // with the current time since boot.
  if (connected) {
    String newValue = "Time since boot: " + String(millis()/1000);
    //Serial.println("Setting new characteristic value to \"" + newValue + "\"");

    // Set the characteristic's value to be the array of bytes that is actually a
string.
    pRemoteCharacteristic->writeValue(newValue.c_str(), newValue.length());
  }else if(doScan){
    BLEDevice::getScan()->start(0);  // this is just eample to start scan after
disconnect, most likely there is better way to do it in arduino
  }
  delaytime=500;
  if(heartbeat<50){
     delaytime=2000;
  }

if(heartbeat>50 && heartbeat<60){
    delaytime=1000;
  }
```

```
if(heartbeat>59 && heartbeat<70){
  delaytime=700;
  }

if(heartbeat>69 && heartbeat<80){
  delaytime=400;
  }

if(heartbeat>79 && heartbeat<90){
  delaytime=200;
  }

if(heartbeat>89 && heartbeat<100){
  delaytime=100;
  }

if(heartbeat>99){
  delaytime=25;
  }

   delay(delaytime);

  // Serial.println("beat delay");
  //delay(heartbeat);
  //Serial.println(delaytime);

//turn on LED L1
  pinMode(LED_1, OUTPUT);     //row 1
  digitalWrite(LED_1, HIGH);
  pinMode(LED_2, INPUT);     //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, OUTPUT);      //row 3
  digitalWrite(LED_3, LOW);
  pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);

  delay(delaytime);

  //turn on LED L2
  pinMode(LED_1, INPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, OUTPUT);     //row 2
  digitalWrite(LED_2, HIGH);
  pinMode(LED_3, OUTPUT);      //row 3
  digitalWrite(LED_3, LOW);
  pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);

   delay(delaytime);

  //turn on LED L3
  pinMode(LED_1, OUTPUT);      //row 1
  digitalWrite(LED_1, HIGH);
  pinMode(LED_2, OUTPUT);     //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, INPUT);     //row 3
  digitalWrite(LED_3, LOW);
    pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);
```

```
  delay(delaytime);

  //turn on LED L4
  pinMode(LED_1, INPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, OUTPUT);    //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, OUTPUT);    //row 3
  digitalWrite(LED_3, HIGH);
    pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);

   delay(delaytime);

//turn on LED L5
  pinMode(LED_1, OUTPUT);     //row 1
  digitalWrite(LED_1, HIGH);
  pinMode(LED_2, OUTPUT);    //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, OUTPUT);    //row 3
  digitalWrite(LED_3, LOW);
  pinMode(LED_4, INPUT);      //row 4
  digitalWrite(LED_4, LOW);

  delay(delaytime);

  //turn on LED L6
  pinMode(LED_1, INPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, OUTPUT);    //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, OUTPUT);    //row 3
  digitalWrite(LED_3, LOW);
  pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, HIGH);

  //turn on LED L7
  pinMode(LED_1, OUTPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, OUTPUT);    //row 2
  digitalWrite(LED_2, HIGH);
  pinMode(LED_3, INPUT);    //row 3
  digitalWrite(LED_3, LOW);
  pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);

  //turn on LED L8
  pinMode(LED_1, OUTPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, INPUT);    //row 2
  digitalWrite(LED_2, LOW);
  pinMode(LED_3, OUTPUT);    //row 3
  digitalWrite(LED_3, HIGH);
  pinMode(LED_4, OUTPUT);      //row 4
  digitalWrite(LED_4, LOW);

  //turn on LED L9
  pinMode(LED_1, OUTPUT);     //row 1
  digitalWrite(LED_1, LOW);
  pinMode(LED_2, OUTPUT);    //row 2
  digitalWrite(LED_2, HIGH);
```

```
pinMode(LED_3, OUTPUT);    //row 3
digitalWrite(LED_3, LOW);
pinMode(LED_4, INPUT);      //row 4
digitalWrite(LED_4, LOW);

//turn on LED L10
pinMode(LED_1, OUTPUT);     //row 1
digitalWrite(LED_1, LOW);
pinMode(LED_2, INPUT);    //row 2
digitalWrite(LED_2, LOW);
pinMode(LED_3, OUTPUT);    //row 3
digitalWrite(LED_3, LOW);
pinMode(LED_4, OUTPUT);      //row 4
digitalWrite(LED_4, HIGH);

//turn on LED L11
pinMode(LED_1, OUTPUT);     //row 1
digitalWrite(LED_1, LOW);
pinMode(LED_2, OUTPUT);    //row 2
digitalWrite(LED_2, LOW);
pinMode(LED_3, OUTPUT);    //row 3
digitalWrite(LED_3, HIGH);
pinMode(LED_4, INPUT);      //row 4
digitalWrite(LED_4, LOW);

//turn on LED L12
pinMode(LED_1, OUTPUT);     //row 1
digitalWrite(LED_1, LOW);
pinMode(LED_2, OUTPUT);    //row 2
digitalWrite(LED_2, LOW);
pinMode(LED_3, INPUT);    //row 3
digitalWrite(LED_3, LOW);
pinMode(LED_4, OUTPUT);      //row 4
digitalWrite(LED_4, HIGH);

}
```

Appendix D: Wifi/API networking and servo control prototype code

```
#include <Arduino.h>

#include <WiFi.h>
#include <WiFiMulti.h>

#include <HTTPClient.h>

#include <WiFiClientSecure.h>

#include <ArduinoJson.h>
#include <Servo.h>

const char* ssid = "opp";
const char* password = "123456789";

static const int servoPin = 4;

JsonObject obj;
DynamicJsonDocument doc(1024);
DeserializationError error;
Servo fitbit_servo;

int time_sleep;

const char* rootCACertificate = \
"-----BEGIN CERTIFICATE-----\n" \
CERTIFICATE REDACTED
"-----END CERTIFICATE-----\n" ;

// Not sure if WiFiClientSecure checks the validity date of the certificate.
// Setting clock just to be sure...
void setClock() {
  configTime(0, 0, "pool.ntp.org", "time.nist.gov");

  Serial.print(F("Waiting for NTP time sync: "));
  time_t nowSecs = time(nullptr);
  while (nowSecs < 8 * 3600 * 2) {
    delay(500);
    Serial.print(F("."));
    yield();
    nowSecs = time(nullptr);
  }

  Serial.println();
  struct tm timeinfo;
  gmtime_r(&nowSecs, &timeinfo);
  Serial.print(F("Current time: "));
  Serial.print(asctime(&timeinfo));
}

void setup() {

    Serial.begin(115200);

    fitbit_servo.attach(servoPin);
```

```
    Serial.println();
    Serial.println();
    Serial.println();

    WiFi.begin(ssid, password);

    while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());

    setClock();

}

void loop() {
  WiFiClientSecure *client = new WiFiClientSecure;
  if(client) {
    client -> setCACert(rootCACertificate);

    {
      HTTPClient https;

      Serial.print("[HTTPS] begin...\n");
      if (https.begin("https://api.fitbit.com/1/user/-/sleep/date/today.json")) {  //
HTTPS
        https.addHeader("Authorization", "redacted");

        Serial.print("[HTTPS] GET...\n");

        int httpCode = https.GET();

        if (httpCode > 0) {
          Serial.printf("[HTTPS] GET... code: %d\n", httpCode);

          if (httpCode == HTTP_CODE_OK || httpCode == HTTP_CODE_MOVED_PERMANENTLY) {
            String payload = https.getString();
            Serial.println(payload);
            error = deserializeJson(doc, payload);
            if (error) {
                Serial.print(F("deserializeJson() failed: "));
                Serial.println(error.c_str());
                return;
            }
            obj = doc.as<JsonObject>();
            Serial.println(F("Response:"));
            Serial.println(doc["totalMinutesAsleep"].as<int>());
            Serial.println(doc["totalSleepRecords"].as<int>());
            Serial.println(doc["totalTimeInBed"].as<int>());

            Serial.print(F("Operate the servo , move to : "));

            if(time_sleep > 480)
            {
              fitbit_servo.write(180);
              Serial.println("180 degrees");
```

```
            }
            else if(480 > time_sleep >= 420)
            {
              fitbit_servo.write(150);
              Serial.println("150 degrees");
            }
            else if(360 > time_sleep >= 300)
            {
              fitbit_servo.write(120);
              Serial.println("120 degrees");
            }
            else if(300 > time_sleep >= 240)
            {
              fitbit_servo.write(90);
              Serial.println("90 degrees");
            }
            else if(time_sleep < 240 )
            {
              fitbit_servo.write(60);
              Serial.println("60 degrees");
            }
          }
        } else {
          Serial.printf("[HTTPS] GET... failed, error: %s\n",
https.errorToString(httpCode).c_str());
        }

        https.end();
      } else {
        Serial.printf("[HTTPS] Unable to connect\n");
      }

    }

    delete client;
  } else {
    Serial.println("Unable to create client");
  }

//  for(int posDegrees = 0; posDegrees <= 180; posDegrees++) {
//        fitbit_servo.write(posDegrees);
//        //Serial.println(posDegrees);
//        //delay(20);
//   }
//
//   for(int posDegrees = 180; posDegrees >= 0; posDegrees--) {
//        fitbit_servo.write(posDegrees);
//        //Serial.println(posDegrees);
//        //delay(20);
//   }

  Serial.println();
  Serial.println("Waiting 60s before the next round...");
  delay(60000);
}
```